



**Universidad de Valladolid**



**ESCUELA DE INGENIERÍAS  
INDUSTRIALES**

**UNIVERSIDAD DE VALLADOLID**

**ESCUELA DE INGENIERIAS INDUSTRIALES**

**Grado en Ingeniería Electrónica y Automática**

**SENSORIZACIÓN PARA ENTORNOS  
COLABORATIVOS**

**Autor:**

**Cuadrado Peñas, Daniel**

**Tutor:**

**González Sánchez, José Luis  
Ingeniería de Sistemas y Automática**

**Valladolid, noviembre 2017.**



Universidad de Valladolid





Universidad de Valladolid





## **AGRADECIMIENTOS**

*Agradecer a mi familia, porque sin ellos no podría haber llegado hasta aquí.*

*También a mis tutores de la Universidad y de Cartif, José Luis González Sánchez y Fernando Gayubo Rojo, por todo el tiempo que han dedicado en mi Trabajo de Fin de Grado, por su ayuda y consejos.*

*A mis compañeros de la División de Mecánica de Cartif, que me han ayudado en todo cuanto he necesitado.*



Universidad de Valladolid





## **RESUMEN**

El Trabajo Fin de Grado que se va a exponer a continuación consiste en el desarrollo de un entorno colaborativo utilizando distintos tipos de sensores, ya sean ultrasonidos, de fuerza o de contacto y la plataforma electrónica ARDUINO MEGA 2560 como cerebro del proyecto.

Para la simulación del movimiento del robot y del objeto o persona con la que va a interactuar el mismo, se van a utilizar dos cintas transportadoras con un motor de corriente continua cada uno.

## **PALABRAS CLAVE**

Robot colaborativo, sensorización, Arduino, cinta transportadora, impresión 3D.



Universidad de Valladolid





### **ABSTRACT**

The project that is going to be exposed below consists of the development of a collaborative environment using different types of sensors, ultrasonic sensors, forced sensors or contact sensors, and the ARDUINO MEGA 2560 as the brain of the project.

Two conveyor belts with a direct current motor each one will be used to simulate the movement of the robot and the object or person with which it will interact.

### **KEYWORDS**

Collaborative robot, sensorization, Arduino, conveyor belt, 3D printing.



Universidad de Valladolid





# ÍNDICE

INTRODUCCIÓN .....	19
Y OBJETIVOS .....	19
1.1 INTRODUCCIÓN .....	21
1.2 OBJETIVOS .....	21
1.3 FASES DEL PROYECTO .....	22
ESTADO DEL ARTE .....	23
2.1 ROBÓTICA COLABORATIVA .....	25
2.2 VENTAJAS .....	27
2.3 INCONVENIENTES .....	28
2.4 EVOLUCIÓN DE LA ROBÓTICA COLABORATIVA .....	28
2.5 MARCAS .....	30
NORMA .....	33
ISO/TS 15066:2016 .....	33
3.1 NORMATIVA .....	35
3.2 PARADA SEGURA MONITORIZADA .....	36
3.3 GUIADO MANUAL .....	36
3.4 CONTROL DE VELOCIDAD Y SEPARACIÓN .....	37
3.5 LIMITACIÓN DE FUERZA Y ENERGÍA POR MEDIO DE FUNCIONES DE MANDO .....	38
SENSORIZACIÓN .....	39
4.1 SENSORIZACIÓN .....	41
4.2 TIPOS DE SENSORES .....	41
4.3 SENSORES DE CONTACTO .....	41
4.3.1 Introducción .....	41
4.3.2 Tipos de sensores de contacto .....	41
4.3.3 Aplicaciones .....	43
4.4 SENSORES DE FUERZA .....	43
4.4.1 Introducción .....	43
4.4.2 Tipos de sensores de fuerza .....	43
4.5 SENSORES DE PROXIMIDAD .....	45
4.5.1 Introducción .....	45
4.5.2 Sensor óptico .....	45



4.5.3	Sensores inductivos.....	49
4.5.4	Sensor capacitivo.....	51
4.5.5	Sensores ultrasónicos.....	53
4.6	SENSORES WEARABLES .....	54
4.6.1	Introducción .....	54
4.6.2	Tecnología wearable en la industria .....	54
4.7	SISTEMA SENSORIAL.....	56
4.7.1	Introducción .....	56
4.7.2	Capacitive touch sensor switch .....	56
4.7.3	HC-SR04 Ultrasonic sensor .....	57
4.7.4	Sensor de fuerza resistivo .....	59
ARDUINO .....		61
5.1	ARDUINO .....	63
5.2	HISTORIA .....	63
5.3	¿POR QUÉ ARDUINO? .....	64
5.4	ARDUINO MEGA 2560.....	65
IMPRESIÓN 3D.....		67
6.1	INTRODUCCIÓN .....	69
6.2	HISTORIA .....	69
6.3	TIPOS DE IMPRESIÓN 3D .....	69
6.4	PRUSA I3.....	72
SOFTWARE .....		73
7.1	SOFTWARE ARDUINO.....	75
7.1.1	FUNCIONES BASICAS .....	76
7.2	CATIA.....	77
7.3	CURA.....	78
DISEÑO MECÁNICO .....		81
8.1	INTRODUCCIÓN.....	83
8.2	DISEÑO .....	83
8.2.1	Banda.....	83
8.2.2	Rodillo.....	84
8.2.3	Tensor.....	85
8.2.4	Motores.....	86
8.2.5	Piezas periféricas.....	87
8.3	PROBLEMAS SURGIDOS.....	89



CONEXIÓN CON ARDUINO .....	91
9.1 INTRODUCCIÓN .....	93
9.2 PROTOBOARD.....	93
9.3 CABLES.....	93
9.4 ARDUINO .....	94
9.4.1 Alarma .....	94
9.4.2 Ultrasonidos .....	95
9.4.3 Touchless.....	95
9.4.4 Sensor resistivo de fuerza .....	96
9.4.5 Motores.....	97
9.4.6 Completo.....	99
ENSAYO DE EXPERIMENTOS.....	101
10.1 INTRODUCCIÓN .....	103
10.2 CALIBRACION DE LOS SENSORES.....	103
10.3 PRUEBA DE EXPERIMENTOS.....	106
10.3.1 Control de velocidad y separación .....	107
10.3.2 Limitación de fuerza y energía .....	108
10.4 CONCLUSIONES .....	109
ESTUDIO ECONÓMICO .....	111
11.1 INTRODUCCIÓN.....	113
11.2 COSTES DIRECTOS .....	113
12.2.1 Costes del personal .....	113
12.2.2 Costes de amortización de hardware y software.....	114
12.2.3 Costes de material de oficina .....	116
12.2.4 Costes directos totales .....	116
11.3 COSTES INDIRECTOS.....	117
11.4 COSTES TOTALES .....	117
CONCLUSIONES Y LÍNEAS FUTURAS .....	119
12.1 CONCLUSIONES .....	121
12.2 LÍNEAS FUTURAS .....	122
BIBLIOGRAFÍA.....	123
13.1 REFERENCIAS RELACIONADAS CON LA ROBOTICA COLABORATIVA .....	125
13.2 REFERENCIAS RELACIONADAS CON LA SENSORIZACIÓN .....	125
13.3 REFERENCIAS RELACIONADAS CON ARDUINO E IMPRESIÓN 3D .....	126
ANEXOS .....	127



I.	PROGRAMACION SOFTWARE ARDUINO .....	129
II.	DATASHEET.....	137



# ÍNDICE DE ILUSTRACIONES

ILUSTRACIÓN 1: ROBOT COLABORATIVO YUMI, ABB.....	25
ILUSTRACIÓN 2: UN GRUPO DE ROBOTS, EN LA LÍNEA DE MONTAJE DE UN VOLKSWAGEN .....	27
ILUSTRACIÓN 3: EVOLUCIÓN DE LOS ROBOTS COLABORATIVOS.....	29
ILUSTRACIÓN 4: ROBOT COLABORATIVO DE ABB.....	30
ILUSTRACIÓN 5: ROBOT COLABORATIVO DE FANUC.....	30
ILUSTRACIÓN 6: ROBOT COLABORATIVO DE KINOVA .....	31
ILUSTRACIÓN 7: ROBOT COLABORATIVO DE KUKA .....	31
ILUSTRACIÓN 8: ROBOT COLABORATIVO DE RETHINK ROBOTICS.....	31
ILUSTRACIÓN 9: ROBOTS COLABORATIVOS DE UNIVERSAL ROBOTS.....	32
ILUSTRACIÓN 10: MODO PARADA DE SEGURIDAD MONITORIZADA .....	36
ILUSTRACIÓN 11: MODO GUIADO MANUAL.....	37
ILUSTRACIÓN 12: MODO DE CONTROL DE VELOCIDAD Y DISTANCIA .....	38
ILUSTRACIÓN 13: MODO DE LIMITACIÓN DE FUERZA Y ENERGÍA.....	38
ILUSTRACIÓN 14: SENSOR DE CONTACTO (PULSADOR) .....	42
ILUSTRACIÓN 15: SENSOR DE CONTACTO (BUMPERS) .....	42
ILUSTRACIÓN 16: MUÑECA DETECTORA DE FUERZA .....	43
ILUSTRACIÓN 17: SENSOR DE ARRAY TÁCTIL .....	44
ILUSTRACIÓN 18: DIAGRAMA BÁSICO DE UN SENSOR ÓPTICO .....	45
ILUSTRACIÓN 19: SENSOR ÓPTICO .....	46
ILUSTRACIÓN 20: SENSOR ÓPTICO DE BARRERA.....	46
ILUSTRACIÓN 21: SENSOR ÓPTICO AUTORREFLEXIVO.....	47
ILUSTRACIÓN 22: SENSOR ÓPTICO REFLEXIVO CON REFLECTOR.....	48
ILUSTRACIÓN 23: DETECCIÓN DEL PASO DE BOTELLAS Y NIVEL DE LÍQUIDO EN UNA AMPOLLA.....	48
ILUSTRACIÓN 24: DETECCIÓN DE TORNILLOS Y VIGILANCIA DE UNA ZONA. ....	49
ILUSTRACIÓN 25: CONTROL DE LA PUERTA DE UN GARAJE Y DETECCIÓN DE OBJETOS EN UNA LÍNEA DE TRANSPORTE. ....	49
ILUSTRACIÓN 26: DIAGRAMA BÁSICO DE UN SENSOR INDUCTIVO .....	50
ILUSTRACIÓN 27: APLICACIONES DE LOS SENSORES INDUCTIVOS .....	51
ILUSTRACIÓN 28: DIAGRAMA BÁSICO DE UN SENSOR CAPACITIVO .....	51



ILUSTRACIÓN 29: SENSOR CAPACITIVO .....	52
ILUSTRACIÓN 30: FUNCIONAMIENTO DE UN SENSOR ULTRASÓNICO.....	53
ILUSTRACIÓN 31: SENSORES WEARABLES .....	55
ILUSTRACIÓN 32: SENSOR CAPACITIVO VMA 305.....	56
ILUSTRACIÓN 33: SENSOR ULTRASÓNICO HC-SR04.....	57
ILUSTRACIÓN 34: FUNCIONAMIENTO DEL SENSOR ULTRASÓNICO HC-SR04.....	58
ILUSTRACIÓN 35: RANGO DE FUNCIONAMIENTO DEL SENSOR ULTRASÓNICO HC-SR04.....	58
ILUSTRACIÓN 36: SENSOR RESISTIVO DE FUERZA .....	59
ILUSTRACIÓN 37: RESISTENCIA VS FUERZA .....	60
ILUSTRACIÓN 38: ARDUINO MEGA 2560 .....	65
ILUSTRACIÓN 39: PIEZA REALIZADA CON IMPRESIÓN 3D.....	69
ILUSTRACIÓN 40: MODELADO POR DEPOSICIÓN FUNDIDA.....	70
ILUSTRACIÓN 41: SINTERIZADO SELECTIVO POR LÁSER.....	71
ILUSTRACIÓN 42: ESTEREOLITOGRAFÍA .....	71
ILUSTRACIÓN 43: IMPRESORA PRUSA I3 .....	72
ILUSTRACIÓN 44: INTERFAZ DEL SOFTWARE DE ARDUINO .....	75
ILUSTRACIÓN 45: INTERFAZ DEL SOFTWARE CATIA .....	78
ILUSTRACIÓN 46: INTERFAZ DEL SOFTWARE CURA.....	79
ILUSTRACIÓN 47: BANDA DE LA CINTA TRANSPORTADORA .....	84
ILUSTRACIÓN 48: RODILLO EN EL SOFTWARE CURA .....	84
ILUSTRACIÓN 49: RODILLO DESPUÉS DE SER IMPRESO .....	85
ILUSTRACIÓN 50: TENSOR DISEÑADO PARA LA CINTA TRANSPORTADORA.....	85
ILUSTRACIÓN 51: MOTOR FAULHABER DE LA PRIMERA CINTA TRANSPORTADORA .....	86
ILUSTRACIÓN 52: MOTOR TERCESA DE LA SEGUNDA CINTA TRANSPORTADORA.....	86
ILUSTRACIÓN 53: PARTE DELANTERA DEL SOPORTE DE SENSORES .....	87
ILUSTRACIÓN 54: PARTE TRASERA DEL SOPORTE DE SENSORES.....	88
ILUSTRACIÓN 55: PLACA SENSADA Y LOS DETECTORES DE CONTACTO .....	88
ILUSTRACIÓN 56: CONTRAPESO DE LA PLACA DE ALUMINIO.....	89
ILUSTRACIÓN 57: MOTOR CON CAJA REDUCTORA.....	90
ILUSTRACIÓN 58: REDUCTORA DEL MOTOR TERCESA .....	90
ILUSTRACIÓN 59: PROTOBOARD.....	93
ILUSTRACIÓN 60: CABLES EMPLEADOS PARA LA CONEXIÓN CON ARDUINO.....	94
ILUSTRACIÓN 61: CONEXIONES ALARMA.....	95



ILUSTRACIÓN 62: CONEXIONES ULTRASONIDOS.....	95
ILUSTRACIÓN 63: CONEXIONES TOUCHLESS.....	96
ILUSTRACIÓN 64: CONEXIÓN FSR.....	97
ILUSTRACIÓN 65: INVERSIÓN DE GIRO DE LOS MOTORES.....	97
ILUSTRACIÓN 66: ESQUEMA DE CONEXIÓN DEL INTEGRADO L293B.....	98
ILUSTRACIÓN 67: CONEXIÓN DE LOS MOTORES Y EL INTEGRADO L293B CON ARDUINO.....	99
ILUSTRACIÓN 68: VISIÓN DE LAS CONEXIONES ENTRE EL PROTOBOARD Y ARDUINO.....	99
ILUSTRACIÓN 69: VISIÓN GENERAL DE LAS CONEXIONES.....	100
ILUSTRACIÓN 70: PROGRAMA SENSOR ULTRASÓNICO.....	103
ILUSTRACIÓN 71: PROGRAMA SENSOR DE CONTACTO DIGITAL.....	104
ILUSTRACIÓN 72: PROGRAMA SENSOR DE CONTACTO ANALÓGICO.....	105
ILUSTRACIÓN 73: PROGRAMA SENSOR RESISTIVO DE FUERZA.....	106
ILUSTRACIÓN 74: CONTROL DE VELOCIDAD Y SEPARACIÓN.....	107
ILUSTRACIÓN 75: LIMITACIÓN DE FUERZA Y ENERGÍA.....	108



Universidad de Valladolid





# ÍNDICE DE TABLAS

TABLA 1: DÍAS LABORABLES EN UN AÑO .....	113
TABLA 2: HORAS DEDICADAS AL PROYECTO .....	114
TABLA 3: COSTES AMORTIZACIÓN DEL MATERIAL DE INFORMÁTICA .....	115
TABLA 4: COSTES DE MATERIAL DE OFICINA .....	116
TABLA 5: TOTAL COSTES DIRECTOS .....	116
TABLA 6: TOTAL COSTES INDIRECTOS .....	117
TABLA 7: COSTE TOTAL DEL PROYECTO .....	117



Universidad de Valladolid





# ***INTRODUCCIÓN Y OBJETIVOS***



Universidad de Valladolid





## 1.1 INTRODUCCIÓN

Estamos ante una nueva Revolución Industrial donde surgen conceptos como Industria 4.0, Internet de las cosas o Big Data. La idea central de esta nueva Revolución es optimizar la producción industrial integrando en un mismo elemento el mundo digital y el real. Para ello se emplea una completa digitalización de procesos en red, innovadores modelos de negocio, nuevos procesos y materiales que harán posible la fabricación de productos de una manera mucho más flexible, eficiente en energía, ahorro de recursos y con un alto nivel de personalización.

El rápido avance de la robótica va a cambiar de forma permanente el mundo, así como Internet ya lo ha hecho. Hoy en día, los robots son un elemento clave de la Industria 4.0, proporcionan respuestas, con nuevos métodos de producción, a las principales cuestiones de nuestros tiempos.

## 1.2 OBJETIVOS

El objetivo del trabajo ha sido la implementación de los conocimientos obtenidos durante el Grado en Electrónica Industrial y Automática en la Universidad de Valladolid en el ámbito de la robótica y de forma práctica aplicado al mundo industrial.

Para ello se va a proceder al diseño de un entorno colaborativo sensorizado que simule un robot colaborativo, también conocidos como cobots, que han sido diseñados para trabajar junto a operarios humanos, ayudando a estos en las tareas que resultan más pesadas y de mayor precisión.

El entorno colaborativo va a constar de varios sensores de proximidad, fuerza y contacto para detectar el objeto y la distancia a los propios sensores y dos cintas transportadoras, cada una moviéndose en una dirección para simular el posible alcance entre el robot y el operario. Las cintas tendrán incorporado un motor de corriente continua, controlado por la placa electrónica utilizada en el proyecto (ARDUINO MEGA 2560), reduciendo sus velocidades según la distancia que nos proporcionen los sensores de proximidad, llegando a detenerlos cuando la distancia sea mínima y pueda haber contacto entre el robot y la persona simuladas con las cintas transportadoras.



## 1.3 FASES DEL PROYECTO

Las fases de desarrollo que se han seguido para el proyecto son las siguientes:

1. Recopilación de información acerca de la robótica colaborativa. Historia, artículos, patentes relacionadas con el mundo de la robótica colaborativa.
2. Planteamiento del proyecto a desarrollar, cómo se va a hacer, fases que se van a seguir, materiales y componentes que se van a utilizar.
3. Adquisición de los sensores que se van a utilizar en el proyecto y realizar una calibración de los mismos con el software de Arduino.
4. Se empieza a construir la cinta transportadora, para ello primero se diseñan las piezas en Catia y posteriormente con el archivo que genera el software Cura se imprimen las piezas en 3D.
5. Con todo el material necesario y las piezas impresas se procede al montaje de la cinta transportadora y se piensa el modo de conexión de los sensores y motores con Arduino.
6. Una vez finalizada la construcción de la cinta y con todas las conexiones realizadas, se desarrolla una tabla de experimentos y se hace un test de prueba y error programando con el software de Arduino.
7. Después de comprobar que todo funciona correctamente, se puede dar por finalizado el proyecto.



---

## CAPÍTULO 2

# ***ESTADO DEL ARTE***

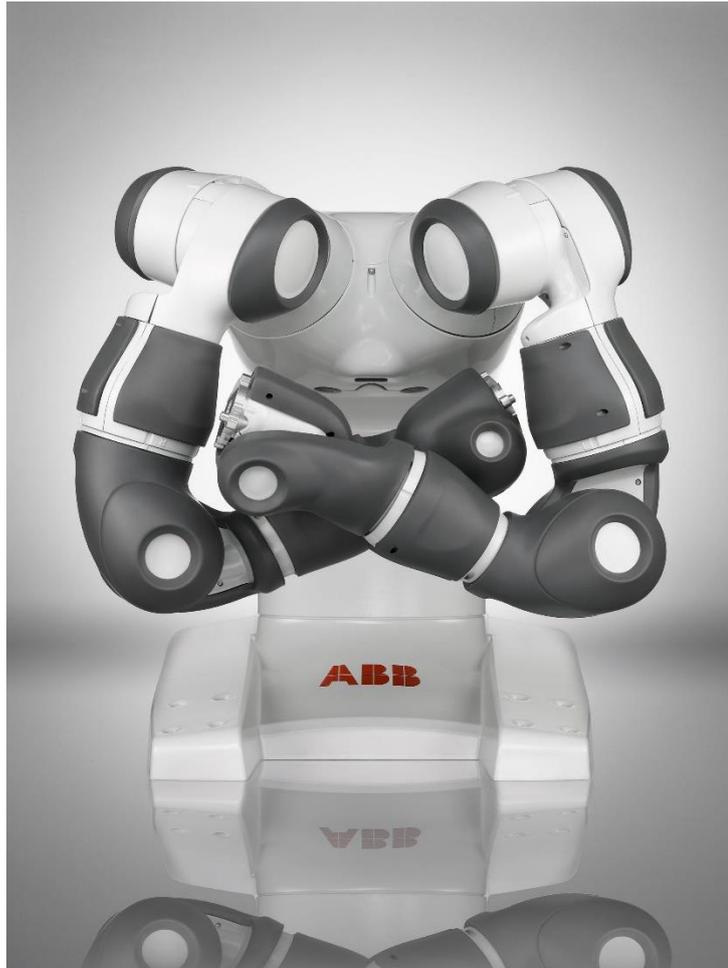


Universidad de Valladolid



## 2.1 ROBÓTICA COLABORATIVA

La robótica colaborativa es un término novedoso en el mundo de la industria y aún con mucha evolución por delante. Los robots colaborativos han sido diseñados para interactuar con el trabajador eliminando los elementos de seguridad que hasta ahora no permitían que robot y operario pudieran “colaborar”.



*Ilustración 1: Robot colaborativo Yumi, ABB*

Para conseguir esta compenetración entre el operario y la máquina, se han tenido que incluir sistemas de sensorización más avanzados, ya sean de visión artificial, sensores de proximidad, sensores de contacto, sensores de presión, etc.

La seguridad debe ser siempre la primera preocupación en cualquier aplicación robótica. Los países tienen diferentes leyes y normas al respecto y algunas compañías pueden tener lineamientos aún más estrictos. [1]

Para evitar un posible accidente debido a la velocidad de desplazamiento del brazo robótico, la nueva tecnología ha avanzado y hay nuevos lazos de control que irán



reduciendo la velocidad a medida que una persona entre dentro de su entorno de trabajo, o si contactan contra el operario o un objeto, el robot consigue detenerse al instante y regresar a una posición de seguridad prefijada.

En cuanto a la morfología y el diseño del brazo los materiales han cambiado y ahora están compuestos de materiales acolchados para absorber el choque contra el humano, y de forma cilíndrica evitando aristas que puedan provocar daños y cortes.

Una noticia llamó la atención sobre el tema de la seguridad en las plantas de trabajo y el peligro que los robots no colaborativos pueden tener si los operarios no tienen el cuidado necesario.

“El 2 de julio de 2015, un trabajador de la planta de Volkswagen en Baunatal (Alemania) fue hallado muerto después de que un robot lo agarrara por el pecho y lo aplastara contra una placa metálica.

El técnico, de 21 años, estaba instalando uno de los robots junto a otro trabajador cuando se produjo el incidente. Los accidentes relacionados con robots son muy poco habituales en las plantas de producción occidentales dado que estos están en jaulas de seguridad precisamente para prevenir incidentes en el contacto con los humanos. En este caso, el fallecido estaba dentro de la jaula, mientras que su compañero estaba fuera.

El robot estaba programado para realizar diversas tareas en el proceso de montaje de los vehículos. Normalmente trabajan desde áreas confinadas, cogiendo partes de los coches y manipulándolas.

El portavoz de Volkswagen Heiko Hillwig ha atribuido el suceso a un "error humano" más que a un problema técnico del robot. En este sentido, ha apuntado que el robot no era de los de última generación de máquinas que trabajan codo con codo con los trabajadores en la línea de producción y ha afirmado que el robot en cuestión no sufría problemas técnicos.” [2]



*Ilustración 2: Un grupo de robots, en la línea de montaje de un Volkswagen*

Por desgracia, son estas noticias las que hacen que el mundo de la robótica evolucione hacia un futuro más seguro, donde el entorno de trabajo no sea un lugar peligroso para las personas.

## 2.2 VENTAJAS

Las ventajas de los robots colaborativos respecto a los robots industriales actuales hacen que esta nueva generación sea un avance muy importante en el mundo de la industria.

Los “cobots” como se conoce de una forma más coloquial a los robots colaborativos aventajan a los tradicionales robots industriales en aspectos como el tamaño reducido y la flexibilidad que favorece la posibilidad de operar desde distintos lugares dentro de una línea de producción, optimizando la productividad, su precio es más asequible, siendo de esta manera ideales para la pequeña y mediana empresa.

Por otra parte en el tema de la programación y el montaje no requiere de personas altamente cualificadas para ello gracias a su fácil configuración.

Todas estas ventajas van a permitir que los robots colaborativos amplíen su mundo llegando a sectores donde no habían podido llegar hasta ahora.

## 2.3 INCONVENIENTES

Por contra, los robots colaborativos carecen de la potencia y velocidad que aportan los robots industriales, pero por otra parte no es necesario, porque su principal función es la de trabajar codo con codo con el operario a su mismo nivel de fuerza y velocidad.

El otro inconveniente es el de la carga que pueden soportar, a día de hoy no es más de [10-15Kg], pero en un futuro no muy lejano a la velocidad que va evolucionando el mundo de la tecnología se conseguirá llegar unos números de carga que se aproximen a los que alcanzan los robots industriales.

## 2.4 EVOLUCIÓN DE LA ROBÓTICA COLABORATIVA

El mercado mundial de los robots industriales está experimentando un crecimiento sin precedentes en los últimos años. A medida que las tecnologías de la robotización evolucionan, su uso se extiende cada vez más hacia sectores como la automoción y la electrónica, actualmente considerados como los principales impulsores de su crecimiento. Según el informe anual publicado por la International Federation of Robotics, en 2015 se instalaron más de 240.000 robots industriales en todo el mundo, lo que supone un crecimiento de un 8% respecto al año anterior. El estudio pronostica, además, que este incremento aumentará un 12% anual hasta finales de 2017.

De hecho, en la actualidad, cabe destacar que las ventas de Robots Colaborativos (Cobots) representan sólo el 5% del mercado total de robots. El sector de la Robótica Colaborativa, de acuerdo con varios estudios de mercado, tiene una previsión de crecimiento de más o menos diez veces entre 2015 y 2020, llegando a 1.000 millones de dólares desde los 95 Millones de dólares en el 2014.

Algunas de las mayores compañías de Robots Colaborativos tienen grandes planes de expansión, ya que buscan aumentar su cuota de mercado no sólo dentro de su ámbito, sino dentro del mercado global de robótica.

La previsión de ventas de robots colaborativos excluyendo los 8 principales fabricantes se observa en la siguiente ilustración.

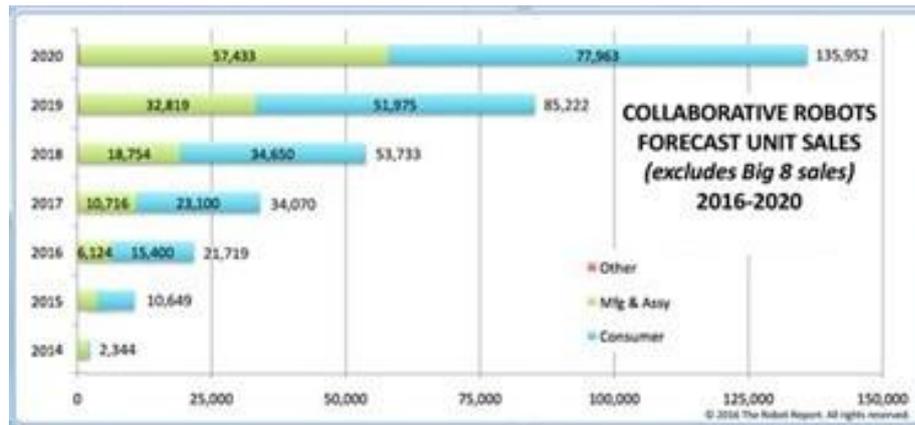


Ilustración 3: Evolución de los robots colaborativos

Lo que comenzó implantándose en pequeñas y medianas empresas se está extendiendo a las empresas de más de 500 empleados.

Entre los casos más significativos de implantación exitosa de Robótica Colaborativa se pueden destacar los siguientes:

BMW, que utiliza aproximadamente 7.500 robots industriales en sus fábricas de todo el mundo, ha experimentado con los Robots Colaborativos llegando a obtener resultados significativos. En un estudio realizado por la marca se demostró que los equipos compuestos de personas y robots que colaboran de manera eficiente, puede ser más productivo que los equipos compuestos de personas o robots solos. También descubrieron que podían aumentar la productividad y reducir las tareas ergonómicamente críticas para las personas, mejorando la capacidad de trabajo de los empleados gracias a estos robots de colaboración fácilmente programables. Según BMW se puede prever duplicar o incluso triplicar el número de robots colaborativos en sus fábricas.

Mercedes Benz, se está desplazando a lo que llama “Robot Farming”: equipar a los trabajadores con una serie de pequeños y más ligeros Cobots o robots colaborativos. La personalización es clave para la comercialización de Mercedes y los compradores están optando por una variedad cada vez mayor de opciones provocando una necesidad de flexibilidad de la que no se dispone actualmente de los robots. Han de ser las personas junto a sus compañeros Cobots los que proporcionen la flexibilidad necesaria. Esto hace pensar que los robots no van a desaparecer por completo, van a ser cada vez más pequeños, más flexibles y operarán en conjunto con los trabajadores en lugar de estar fuera del área de trabajo detrás de las vallas de seguridad.

Whirlpool utiliza robots programables con un solo brazo que han asumido algunas tareas repetitivas, trabajando en concierto con sus colegas humanos. Este robot saca fotografías de los productos para buscar defectos mientras el personal se encarga de revisar y arreglar las conexiones del cableado.

En la fábrica de Whirlpool en Marion, 8 robots colaborativos ayudan a 2.200 empleados a producir alrededor de 18.000 máquinas secadoras al día. Los Cobots,

además de revisar los aparatos para detectar posibles averías, mueven los tambores grandes, instalan los motores y prueban las trampas de pelusa. De esta forma Whirlpool está remodelando sus plantas en torno a los “robots colaborativos” que pueden detener sus labores si una persona interfiere o tropieza con ellos. Esa precaución les permite operar en espacios reducidos con poca o ninguna barrera protectora.

En una planta del fabricante de aviones estadounidense Boeing Co., los robots colaborativos alisan partes remachadas de los aviones 787, algo que pronto podrían hacer en otra de sus fábricas. [3]

## 2.5 MARCAS

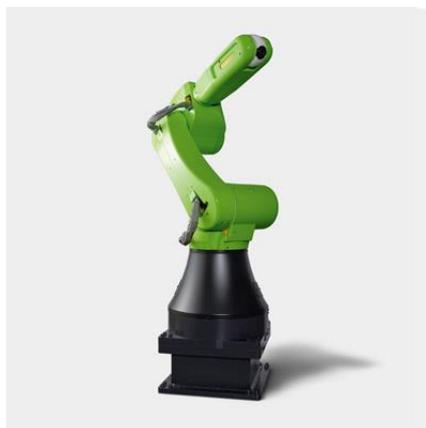
A Continuación se muestran algunas de las principales marcas con Robots Colaborativos: [4]

- ABB



*Ilustración 4: Robot colaborativo de ABB*

- FANUC



*Ilustración 5: Robot colaborativo de FANUC*

- KINOVA



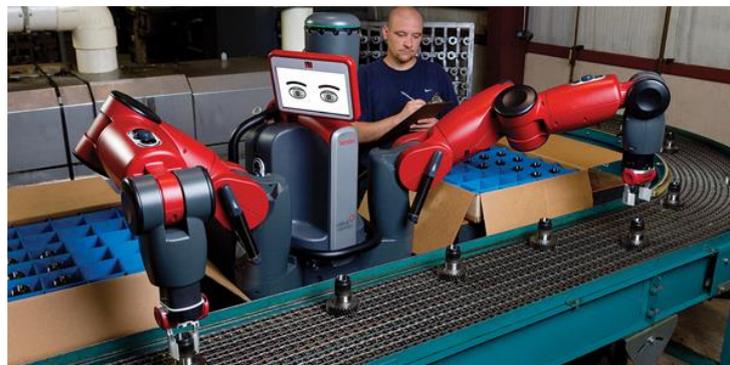
*Ilustración 6: Robot colaborativo de KINOVA*

- KUKA



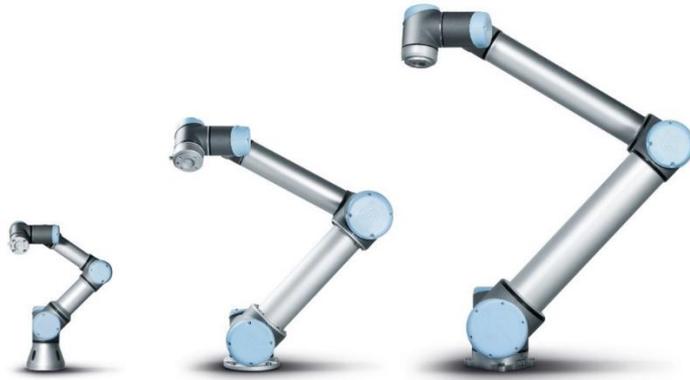
*Ilustración 7: Robot colaborativo de KUKA*

- RETHINK ROBOTICS



*Ilustración 8: Robot colaborativo de RETHINK ROBOTICS*

• UNIVERSAL ROBOTS



*Ilustración 9: Robots colaborativos de UNIVERSAL ROBOTS*



***NORMA***  
***ISO/TS 15066:2016***



Universidad de Valladolid





## 3.1 NORMATIVA

De la misma manera que va evolucionando la robótica, tienen que modificarse las normas de seguridad.

La nueva norma que ISO (Organización Internacional de Normalización) ha publicado, da la indicación para asegurar la integridad física de los trabajadores que operan con robots colaborativos.

Esta norma ISO/TS 15066:2016 - Robots and robotic devices establece los requisitos de seguridad para los sistemas de robot industrial colaborativos. El documento consta de 33 páginas que proporcionaran ayuda a los integradores de celdas robotizadas para llevar a cabo evaluaciones de riesgo al instalar los robots colaborativos.

La norma ISO / TS 15066 comenzó a trabajarse en 2010 y ha sido redactada por un comité de la ISO con miembros procedentes de 24 países, incluidos representantes de los fabricantes de robots colaborativos y no es aplicable a las instalaciones de robots colaborativos que fueron diseñados antes de su publicación.

La especificación ISO proporciona una guía para instalaciones de sistemas donde los robots y las personas comparten el mismo espacio de trabajo. En este tipo de operaciones, la integridad del sistema de control relacionado con la seguridad es de gran importancia, sobre todo cuando se controlan parámetros como la velocidad y la fuerza. Se necesita una evaluación exhaustiva del riesgo para evaluar no sólo el sistema de robot en sí, sino también el entorno en el que se coloca.

La norma describe diferentes conceptos de colaboración y los requisitos necesarios para lograrlos. Además de cubrir el diseño y evaluación de riesgos, también incluye los resultados de un estudio de investigación sobre el umbral del dolor en comparación con la velocidad del robot, la presión y el impacto de partes específicas del cuerpo.

Los suplementos de especificaciones y soporta los estándares de seguridad del robot industriales generales, la norma ISO 10218-1 e ISO 10218-2, y proporciona orientación adicional para los robots colaborativos. Las operaciones de colaboración descritas dependen del uso de robots que cumplen con los requisitos de la norma ISO 10218-1 y se integran para cumplir con la norma ISO 10218-2.

La ISO también señala que la operación de colaboración es un campo en desarrollo, y que los valores de potencia y fuerza de limitación incluyen en la nueva especificación técnica probablemente evolucionará en ediciones futuras. [5]

Esta nueva norma propone 4 tipos de operaciones colaborativas que se pueden dar: [6]

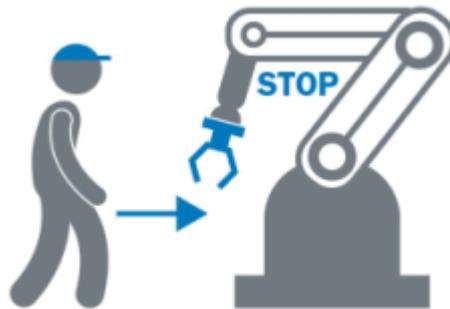
## 3.2 PARADA SEGURA MONITORIZADA

Cuando una persona se encuentre dentro del espacio colaborativo de trabajo, el robot debe detener su marcha y no podrá reiniciarla hasta que dicha persona haya abandonado el espacio.

Es decir el robot o el operador humano se mueve, pero nunca ambos al mismo tiempo, por lo tanto no se puede considerar como un modo de trabajo colaborativo.

Para ello, el sistema de seguridad debe disponer de dispositivos de detección de la presencia del operario, considerando siempre que su localización debe ser coherente con la necesidad de disponer de un tiempo de parada en función de la velocidad de aproximación.

El beneficio que se ofrece es la facilidad y la rapidez para reanudar el movimiento del robot automáticamente.



*Ilustración 10: Modo parada de seguridad monitorizada*

## 3.3 GUIADO MANUAL

En este caso el robot tiene que tener limitación en los ejes y en las velocidades.

Es necesario la detección de personas dentro del espacio colaborativo cuando la trayectoria del robot sea automática.

El robot debe parar y permanecer detenido cuando un operario ingrese en el espacio colaborativo de trabajo, una vez a ocurrido esto, la operación de guiado manual será posible accionando un mando de validación y con una velocidad lo suficientemente baja para que sea seguro.

Se considera una operación automática, no una operación manual.

El robot puede reiniciar su marcha cuando la persona haya abandonado el espacio de trabajo colaborativo.



*Ilustración 11: Modo guiado manual*

### **3.4 CONTROL DE VELOCIDAD Y SEPARACIÓN**

Para que el riesgo sea lo más pequeño posible, la separación mínima entre operario y robot se tiene que mantener, ya sea disminuyendo la velocidad o variando la trayectoria del robot.

Como en el modo de operación anterior, el robot tiene que tener limitación en los ejes y en las velocidades y es necesario la detección de todas las personas dentro del espacio colaborativo.

Si se produce un fallo en el sistema y no se pueda asegurar la mínima distancia entre robot y operario, tiene que iniciarse una parada de seguridad.

Otro factor importante es la velocidad de movimiento, de modo que cuando la velocidad es más pequeña, la distancia de seguridad se puede reducir automáticamente, siempre también de acuerdo a la evaluación de riesgos.

La especificación técnica ISO/TS 15066:2016 facilita criterios técnicos para calcular estas velocidades y distancias.

El robot puede reiniciar su trabajo normal cuando la persona haya abandonado el espacio de trabajo colaborativo.



Ilustración 12: Modo de control de velocidad y distancia

### 3.5 LIMITACIÓN DE FUERZA Y ENERGÍA POR MEDIO DE FUNCIONES DE MANDO

En este modo de trabajo no se trata de evitar el contacto entre robot y operario, el control se basa en reducir el riesgo de accidente ya sea limitando la fuerza o la potencia por debajo de unos valores fijados y así no resulten peligrosos para el trabajador.

Estos valores límite se determinan basándose en las fuerzas y presiones que puede admitir el cuerpo humano sin sufrir lesión alguna. La norma ISO/TS 15066:2016 recoge en su anexo los criterios basados en estudios empíricos.

Para controlar estas fuerzas y presiones se recurre a aumentar las superficies de contacto (redondeadas y sin aristas vivas...), reducir las fuerzas de impacto (acolchados, partes deformables...) y limitar las masas en movimiento y la velocidad (en este caso para reducir la inercia).



Ilustración 13: Modo de limitación de fuerza y energía



# ***SENSORIZACIÓN***



Universidad de Valladolid



## 4.1 SENSORIZACIÓN

La sensorización se podría definir como un medio de captar información del entorno mediante distintos dispositivos que registran eventos físicos, y que a continuación son procesados y gestionados como datos. Esos datos son convertidos, mediante autómatas, sistemas de información o la combinación de ambos, en información que puede proponer soluciones y servicios inteligentes para el ámbito de la industria. En este sentido, la sensorización se inscribe en la llamada Cuarta Revolución Industrial o Industria 4.0, que integra las tecnologías de la información en los procesos productivos en las denominadas fábricas inteligentes. [7]

## 4.2 TIPOS DE SENSORES

Según el funcionamiento de los sensores se pueden clasificar de diversas maneras, en este caso la clasificación que se va a hacer es la siguiente:

- SUPERFICIALES (Sensores de contacto, de fuerza, táctiles, de proximidad).
- EXTERNOS (Visión artificial 2D/3D, wearables).

## 4.3 SENSORES DE CONTACTO

### 4.3.1 Introducción

Los sensores de contacto son los dispositivos más simples de todos los sensores que podemos encontrar en el mercado. Nos indican simplemente si ha habido contacto o no con algún objeto, sin considerar la magnitud de la fuerza de contacto.

Debido a su simplicidad, estos sensores son muy utilizados en la construcción de microrrobots, así como para detectar el final del recorrido o la posición límite de componentes mecánicos.

### 4.3.2 Tipos de sensores de contacto

Se diferencian dos grandes tipos de sensores de contacto que son los más comúnmente utilizados: [8]

### • PULSADOR

Como se aprecia en la imagen siguiente, el funcionamiento de este tipo de sensor es muy sencillo. Está formado por un botón o pulsador que al ser accionado cierra el interruptor interno del sensor indicando que ha habido contacto.

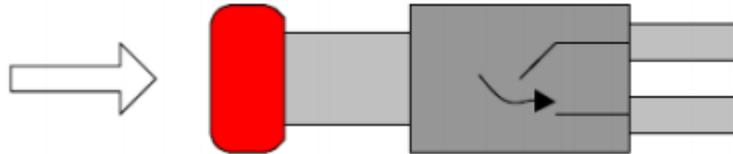


Ilustración 14: Sensor de contacto (Pulsador)

### • FINAL DE CARRERA (BUMPERS)

El bumper es un conmutador con dos posiciones, un muelle de retorno a la posición de reposo y una palanca de accionamiento que varía de tamaño según el modelo seleccionado.

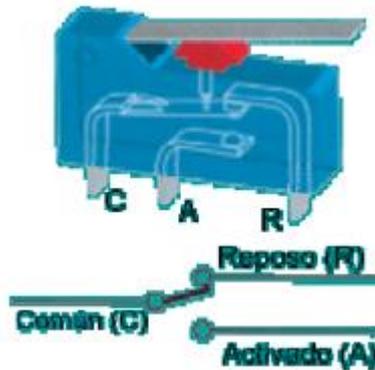


Ilustración 15: Sensor de contacto (Bumpers)

En cuanto a su funcionamiento, también es muy sencillo: en estado de reposo la patilla común (C) y la de reposo (R) se encuentran en contacto permanente hasta que la presión aplicada a la palanca del bumper hace saltar la pequeña pletina interior y entonces el contacto pasa de la posición de reposo a la de activo (A), se puede escuchar cuando el bumper cambia de estado, porque se oye un pequeño clic, esto sucede casi al final del recorrido de la palanca.

### 4.3.3 Aplicaciones

Estos sensores se pueden situar en las pinzas de los brazos de robot para determinar cuando se ha cogido un objeto, pueden formar parte de sondas de inspección para determinar dimensiones de objetos, o incluso pueden situarse en el exterior de las pinzas para ir tanteando un entorno.

Otra aplicación del mismo es saber si se abre o se cierra una puerta completamente y accionar un cambio de dirección del motor o detenerlo.

## 4.4 SENSORES DE FUERZA

### 4.4.1 Introducción

Los sensores de fuerza determinan, Además de si ha habido contacto con un objeto, la magnitud de la fuerza con la que se ha producido dicho contacto. Esta capacidad es muy útil ya que permitirá al robot poder manipular objetos de diferentes tamaños e incluso colocarlos en lugares muy precisos. Para detectar la fuerza con la que se ha contactado con un objeto existen diversas técnicas: [9]

### 4.4.2 Tipos de sensores de fuerza

#### • MUÑECA DETECTORA DE FUERZA

Consta de una célula de carga que se sitúa entre la muñeca y las pinzas del brazo. Su objetivo es proporcionar información sobre las tres componentes de la fuerza ( $F_x, F_y, F_z$ ) y sobre sus tres momentos en velocidad con la que se mueve el brazo es elevada, resulta difícil poder controlar sus movimientos lo suficientemente rápido como para que no provoque ninguna catástrofe (como el aplastamiento de algún objeto).

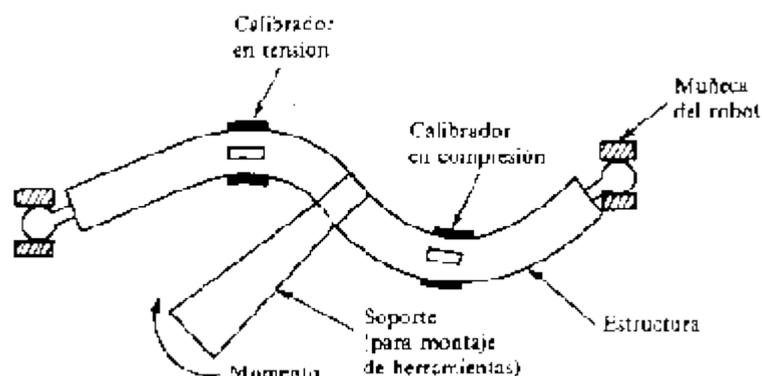


Ilustración 16: Muñeca detectora de fuerza

## • DETECCIÓN DE ARTICULACIONES

Esta técnica se basa en la medida del par de torsión de la articulación. La medida de este par puede resultar sencilla, ya que es proporcional a la corriente que circula por el motor que provoca dicha torsión.

A pesar de que esta técnica pueda parecer sencilla y fiable, tiene un problema importante. La medida del par de torsión se realiza sobre las articulaciones del brazo y no sobre el efector final (la pinza) como sería deseable, por lo que dicha torsión no solo refleja la fuerza que se ejercerá en la pinza, sino también la fuerza utilizada para mover la articulación.

## • SENSORES DE ARRAY TÁCTIL

Es un tipo especial de sensores de fuerza ya que en realidad está constituido por una matriz de pequeños sensores de fuerza. Debido a esta característica, permiten además reconocer formas en los objetos que se está manipulando. Este tipo de dispositivos suelen montarse en las pinzas de los brazos de robot.

Cada uno de los sensores de fuerza que componen la matriz suele ser una almohadilla elastomérica, que cuando se comprime cambia su resistencia eléctrica de manera proporcional a la fuerza aplicada. Midiendo esa resistencia es cuando podemos obtener la información acerca de la fuerza. La resolución de este tipo de sensores vendrá dada lógicamente por las dimensiones de la matriz de sensores.

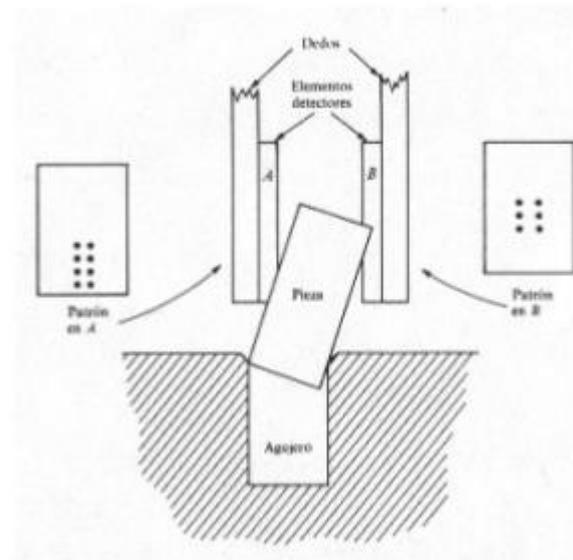


Ilustración 17: Sensor de array táctil

Un factor muy importante y que puede resultar un problema al diseñar este tipo de sensores es el grado de desgaste de la superficie de contacto.

## 4.5 SENSORES DE PROXIMIDAD

### 4.5.1 Introducción

Los sensores de proximidad se utilizan en robótica para la detección de la presencia de objetos (u obstáculos) sin la necesidad de entrar en contacto con ellos y por tanto se utilizan para medir distancias a los mismos. Esta información es de vital importancia en los robots, especialmente si se mueven en entornos no estructurados que requieran de la capacidad para construir mapas del entorno y evitar obstáculos cercanos.

Las distancias de detección dependen básicamente de la tecnología del sensor y del medio de transmisión, y llegan desde pocos milímetros hasta varios metros, con lo que su uso en los robots está condicionado al tipo de aplicación a la que se oriente el robot y la necesidad o no de detectar la presencia de otros objetos del entorno. Dentro de los criterios de selección también se incluyen el tipo de haz que tiene cada sensor (tipo rayo o tipo cono) y la forma en la que rebota la señal sobre los objetos (especular o no especular), influencia del color y/o materiales, repetitividad, precio, entre otros criterios adicionales.

Según el tipo de parámetro variable se pueden distinguir varios tipos de sensores de proximidad:

### 4.5.2 Sensor óptico.

#### 4.5.2.1 Introducción

Un sensor óptico se basa en el aprovechamiento de la interacción entre la luz y la materia para determinar las propiedades de ésta. Una mejora de este tipo de sensores es la utilización de la fibra óptica como elemento de transmisión de la luz.

Los detectores ópticos basan su funcionamiento en la emisión de un haz de luz que es interrumpido o reflejado por el objeto a detectar. Tiene muchas aplicaciones en el ámbito industrial y son ampliamente utilizados.

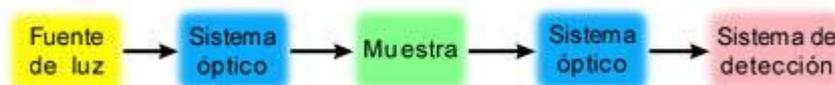


Ilustración 18: Diagrama básico de un sensor óptico

La fuente genera luz de una cierta frecuencia para que el sensor pueda detectarla más fácilmente y diferenciarla de otras fuentes cercanas.

El tipo de luz más usado es el infrarrojo y, en este caso, para diferenciar la luz de la fuente se deja una frecuencia fija y lo que se hace es emitir pulsos. El sensor simplemente descartará cualquier luz infrarroja que le llegue y que no sea intermitente, pudiendo así diferenciar la luz de la fuente de radiación infrarroja proveniente de otros objetos que generen calor.

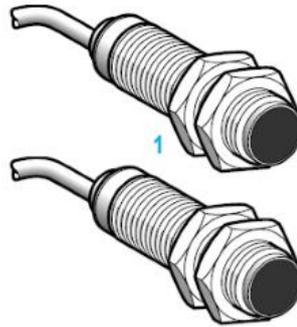


Ilustración 19: Sensor óptico

#### 4.5.2.2 Tipos de sensores ópticos

- **BARRERA**

Para este tipo de sensores ópticos, el sistema emisor y receptor son distintos y están separados pero alineado, estando en frente uno del otro. Es el modo de funcionamiento más fiable de todos, pues toda la potencia que emite el emisor es enviada directamente al receptor, haciendo que la suciedad y la humedad no sean un factor menos negativo para el funcionamiento del sensor.

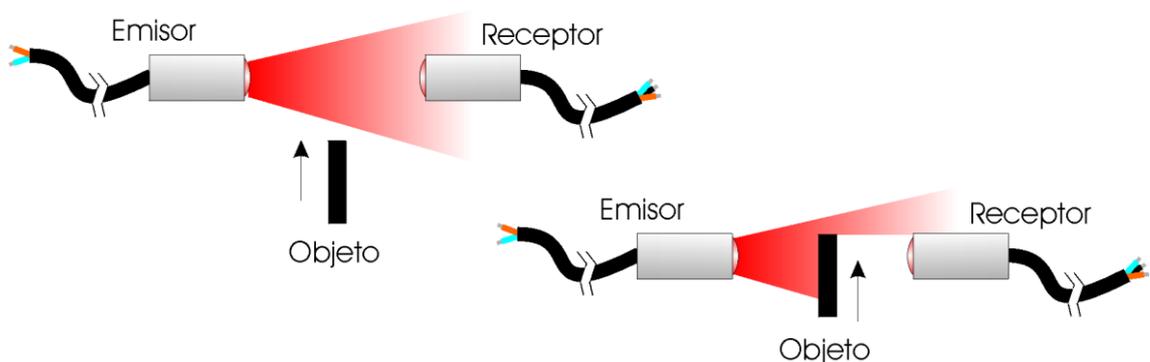


Ilustración 20: Sensor óptico de barrera

Con las fotocélulas de barrera, se consiguen distancias más largas a sensar que con el resto de tipos de funcionamiento, y la distancia entre emisor y receptor no va a depender del color del objeto a detectar.

El mayor inconveniente que presentan es que no consiguen detectar objetos transparentes o translúcidos, la luz que proyecta el emisor atraviesa el cuerpo del objeto y llega al receptor sin detectarlo.

### •AUTORREFLEXIVOS

En este tipo de dispositivos, emisor y receptor forman parte del mismo componente. La luz enviada por el emisor incide sobre el objeto que se quiere detectar y es reflejada, siendo capturada por el receptor.



*Ilustración 21: Sensor óptico autorreflexivo*

Es el tipo de sensor óptico más económico, pero su modo de funcionamiento es poco adecuado en lugares donde la suciedad y la humedad abundante. Ambos factores pueden llegar a “cegar” la fotocélula, haciendo que la detección resulte prácticamente imposible.

La principal ventaja de este tipo de sensores, es que al estar el emisor y el receptor en el mismo encapsulado, sólo se necesita un punto de alimentación y su montaje es rápido y sencillo. Estas fotocélulas se emplean en aplicaciones donde por espacio o accesibilidad, resulta imposible colocar un componente receptor o espejo.

Por otra parte, la distancia de detección que se consigue con las fotocélulas autorreflexivas es más corta. Esta distancia va a depender directamente del color del objeto a detectar, debido a que cada color tiene un factor de reflexión de la luz diferente.

### • REFLEXIVOS CON REFLECTOR

Del mismo modo que los sensores ópticos autorreflexivos, el emisor y el receptor están concentrados en la misma capsula. En estas fotocélulas, la luz que emite el emisor es reflejada por un reflector y detectada por el receptor. Cuando el haz de luz se interrumpe por el objeto que se quiere detectar, la luz no regresa al receptor y se da por detectado el objeto.



Ilustración 22: Sensor óptico reflexivo con reflector

Al tener exclusivamente un encapsulado, sucede lo mismo que en el caso anterior, solamente es necesario alimentar en un único punto, lo que hace más sencillo su montaje en instalación.

La principal ventaja de las fotocélulas reflexivas con reflector es que se consiguen distancias de detección más grandes que con las fotocélulas autorreflexivas. En cambio, estas distancias siguen siendo inferiores a las conseguidas con las fotocélulas de tipo barrera.

En estos casos, la distancia entre la fotocélula y el reflector no va a depender del color del objeto que se quiere detectar. Para que la luz no pueda atravesar el objeto a detectar y se pueda llevar a cabo la detección, se requiere que dicho objeto tenga un cierto grado de opacidad. [10

]

### 4.5.2.3 Aplicaciones

Algunas de las aplicaciones que tienen este tipo de sensores en la industria son las siguientes:

- Detección del paso de botellas y nivel de líquido en una ampolla.

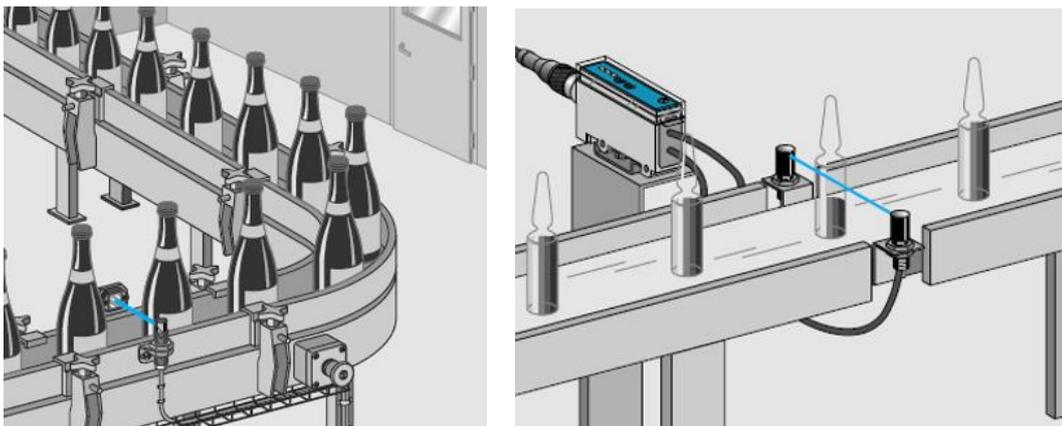


Ilustración 23: Detección del paso de botellas y nivel de líquido en una ampolla.

- Detección de tornillos y vigilancia de una zona.

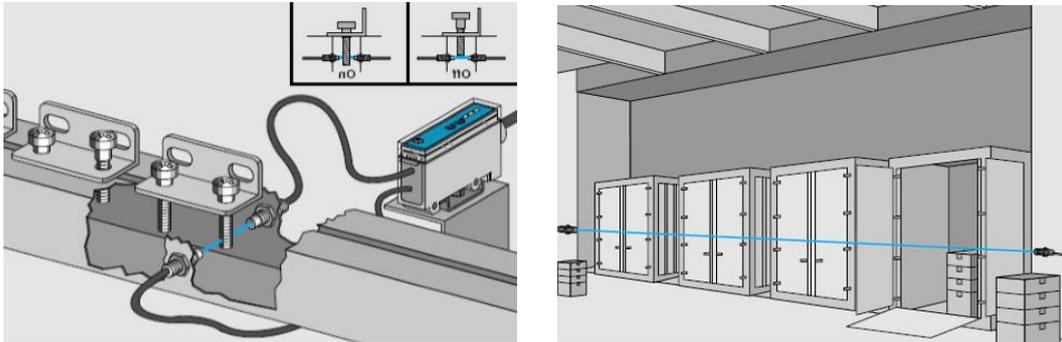


Ilustración 24: Detección de tornillos y vigilancia de una zona.

- Control de la puerta de un garaje y detección de objetos de color oscuro en una línea de transporte.

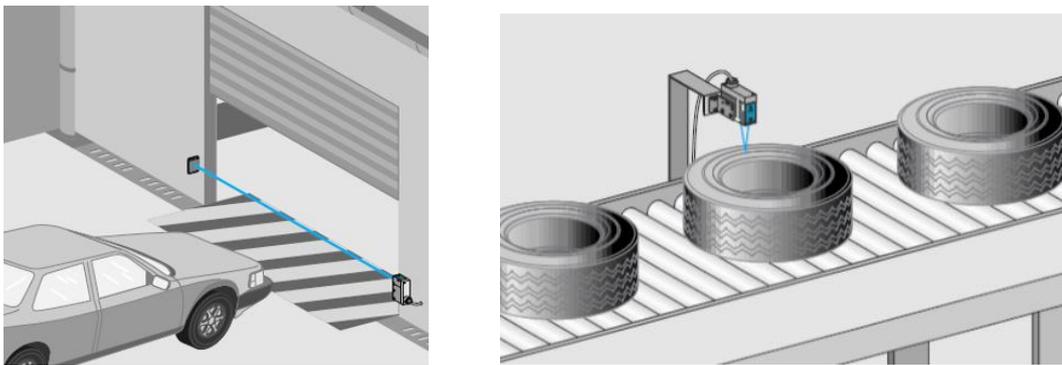


Ilustración 25: Control de la puerta de un garaje y detección de objetos en una línea de transporte.

## 4.5.3 Sensores inductivos

### 4.5.3.1 Introducción

Los sensores inductivos son detectores electrónicos que sirven para detectar materiales metálicos sin necesidad de contacto físico, se basa en la variación de los campos electromagnéticos. Pueden detectar objetos que se le acerquen tanto axial como lateralmente. La distancia máxima a la que son capaces de detectar un objeto está en torno a los 20 milímetros según el tipo y tamaño de este.

En la siguiente imagen se pueden ver los componentes de un sensor inductivo:

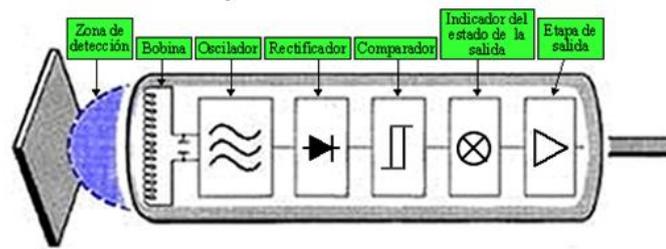


Ilustración 26: Diagrama básico de un sensor inductivo

El principio básico consiste en conseguir el disparo de señal provocado en el comparador que detecta cambios entre la señal emitida por el oscilador y la señal detectada por el circuito de inducción al aproximarse a un cuerpo que provoque cambios en el campo magnético inicial generado por el oscilador. En realidad, al aproximar el campo magnético generado a un cuerpo metálico conductor, se generan a su vez una inducción eléctrica en dicho conductor. Esa tensión provoca la aparición de corrientes internas de Foucault que a su vez generan un campo inducido de respuesta al generado por el detector. El campo resultante es detectado en el comparador y ante un cambio desencadena el proceso de detección. Excitando de esa manera la etapa de salida.

El oscilador crea un campo de alta frecuencia de oscilación por el efecto electromagnético producido por la bobina en la parte frontal del sensor centrado con respecto al eje de la bobina. Así, el oscilador consume una corriente conocida. El núcleo de ferrita concentra y dirige el campo electromagnético en la parte frontal, convirtiéndose en la superficie activa del sensor.

Cuando un objeto metálico interactúa con el campo de alta frecuencia, se inducen corrientes EDDY en la superficie activa. Esto genera una disminución de las líneas de fuerza en el circuito oscilador y, en consecuencia, desciende la amplitud de oscilación.

El circuito detector reconoce un cambio específico en la amplitud y genera una señal, la cual cambia (pilotea) la salida de estado sólido a "ON" u "OFF". Cuando se retira el objeto metálico del área de sensor, el oscilador genera el campo, permitiendo al sensor regresar a su estado normal. [11]

#### 4.5.3.2 Aplicaciones

La principal aplicación de los sensores inductivos es la detección de piezas metálicas. Debido a su funcionamiento, en el que detectan los objetos sin contacto físico: permiten el contaje, analizar su posición forma de objetos metálicos, se pueden emplear en la industria alimentaria, ya que no interfiere en los productos.

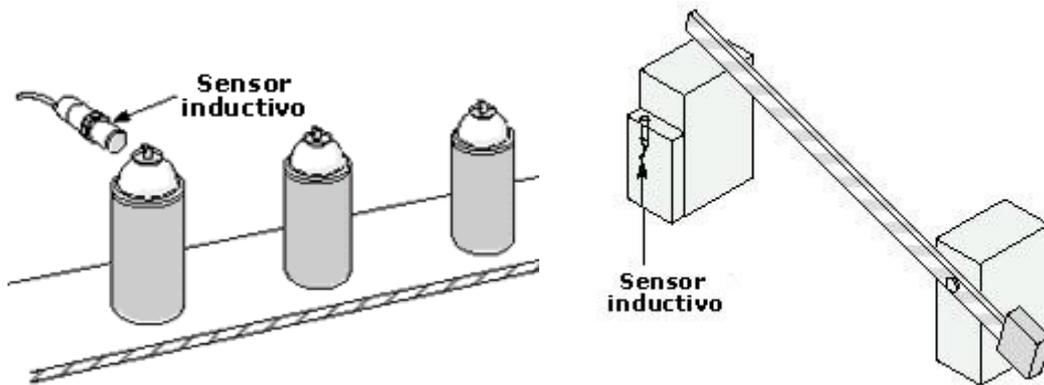


Ilustración 27: Aplicaciones de los sensores inductivos

Este tipo de sensores son ampliamente usados en industrias, como las relacionadas con el automóvil, debido a que la mayoría de las piezas empleadas son metálicas. Los sensores inductivos debido a su elevada conmutación del punto de trabajo se pueden emplear para determinar el sentido de giro y número de vueltas de un engranaje o eje.

## 4.5.4 Sensor capacitivo

### 4.5.4.1 Introducción

Los detectores capacitivos son “interruptores electrónicos” de característica estática que actúan sin elementos electromecánicos. Su funcionamiento se basa en un circuito oscilante RC y las líneas del campo eléctrico que se cierran a través del aire. La aproximación de un objeto con una constante dieléctrica superior a la del aire, ocasiona el desequilibrio del circuito y el inicio de las oscilaciones.

Esta variación de la capacidad es función no sólo de la constante dieléctrica, sino también del volumen, densidad y compacticidad del objeto o sustancia a detectar. Es por esta razón por lo que la mayor parte de detectores capacitivos llevan incorporado un ajuste de sensibilidad para adaptarlo a cada uno de los elementos a detectar de acuerdo con sus propias características y constitución.

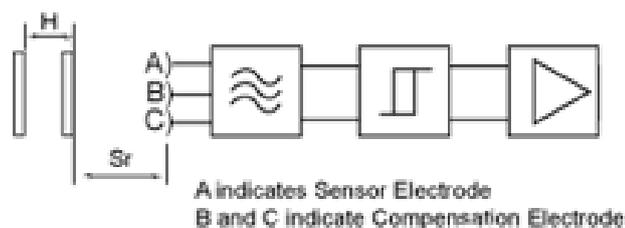


Ilustración 28: Diagrama básico de un sensor capacitivo

#### 4.5.4.2 Características

- Sensibles a la mayoría de líquidos y materiales, permitiendo la detección de otros materiales a través de materiales o paredes no conductores (presencia de agua en el interior de una tubería plástica o envoltorio metálico en el interior de una caja de cartón).
- En general, los líquidos y sólidos conductores son detectados a una distancia mayor que los materiales aislantes, ligeros o porosos. Si la distancia de detección disminuye, la histéresis también disminuye.
- Válidos para materiales no conductores como plástico, cristal, goma y conductivos como metales o agua.
- Versiones para montaje saliente o enrasado (montaje empotrado / no empotrado).
- Las distancias de funcionamiento dependen de la constante dieléctrica del material que se desea detectar. Cuanto mayor sea este valor más fácilmente se detectará el material. [12]

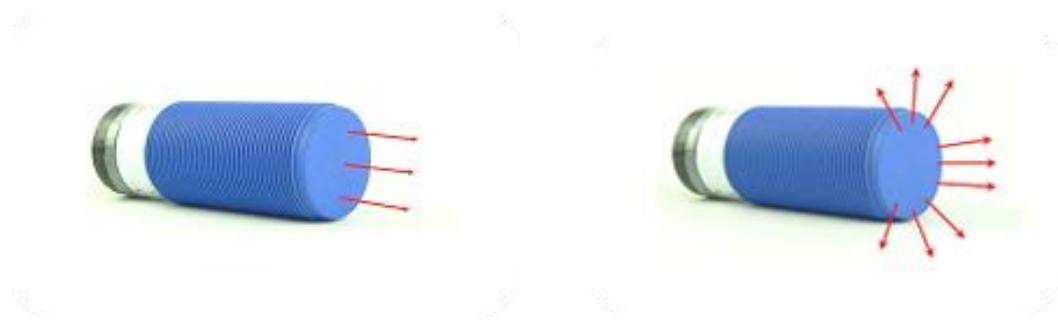


Ilustración 29: Sensor capacitivo

#### 4.5.4.3 Aplicaciones

Estos sensores se emplean para la identificación de objetos, para funciones contadoras y para toda clase de controles de nivel de carga de materiales sólidos o líquidos. También son utilizados para muchos dispositivos con pantalla táctil, como teléfonos móviles, ya que el sensor percibe la pequeña diferencia de potencial entre membranas de los dedos eléctricamente polarizados de una persona.

##### • DETECCIÓN DE NIVEL

En esta aplicación, cuando un objeto (líquidos, granulados, metales, aislantes, etc.) penetra en el campo eléctrico que hay entre las placas sensor, varía el dieléctrico, variando consecuentemente el valor de capacitancia.

### • DETECCIÓN DE HUMEDAD

El principio de funcionamiento de esta aplicación es similar a la anterior. En esta ocasión el dieléctrico, por ejemplo el aire, cambia su permitividad con respecto a la humedad del ambiente.

### • DETECCIÓN DE POSICIÓN

Esta aplicación es básicamente un condensador variable, en el cual una de las placas es móvil, pudiendo de esta manera tener mayor o menor superficie efectiva entre las dos placas, variando también el valor de la capacitancia, y también puede ser usado en industrias químicas. Pero como sabemos este tipo de aplicación no suele ser lo correcto.

## 4.5.5 Sensores ultrasónicos

### 4.5.5.1 Introducción

Los ultrasonidos son sonidos iguales a los que se pueden escuchar, pero con una frecuencia mayor a la máxima que puede percibir el oído humano. Ésta comienza desde unos 16 Hz y tiene un límite superior de aproximadamente 20 KHz, mientras que nosotros vamos a utilizar sonido con una frecuencia de 40 KHz. A este tipo de sonidos es a lo que llamamos Ultrasonidos. El funcionamiento básico de los ultrasonidos como medidores de distancia se muestra de una manera muy clara en el siguiente esquema, donde se tiene un receptor que emite un pulso de ultrasonido que rebota sobre un determinado objeto y la reflexión de ese pulso es detectada por un receptor de ultrasonidos:

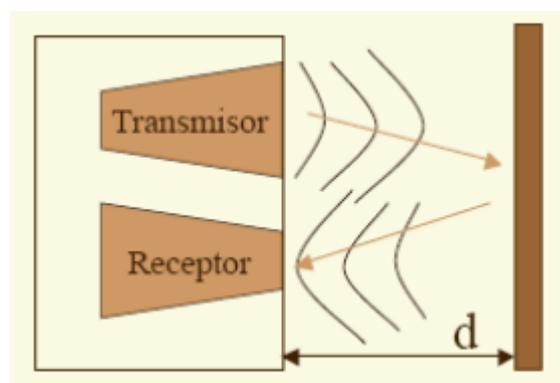


Ilustración 30: Funcionamiento de un sensor ultrasónico

La mayoría de los sensores de ultrasonido de bajo coste se basan en la emisión de un pulso de ultrasonido cuyo campo de acción es de forma cónica. Midiendo el tiempo que transcurre entre la emisión del sonido y la percepción del eco se puede



establecer la distancia a la que se encuentra el obstáculo que ha producido la reflexión de la onda sonora, mediante la fórmula: donde  $V$  es la velocidad del sonido en el aire y  $t$  es el tiempo transcurrido entre la emisión y recepción del pulso.

Existe una línea versátil de sensores que incluyen 30 mm de laminilla metal y albergues plásticos en dos estilos de albergue rectangulares.

Es estrecho análogo y con rendimientos a dispositivos discretos extensamente, sensor múltiple de posicionamiento sensando los rasgos ambientales del entorno del robot.

La mayoría de los robots industriales se limitan a ejecutar de forma repetida secuencias de movimientos fijas que asumen un entorno productivo estructurado y en el que no se producen incidencias. La incorporación de sensores como sistemas de visión y sensores de fuerza distribuidos por el brazo robot, hará que estos sean más flexibles, adaptándose a entornos y flujos de producción menos estructurados y más reactivos en situaciones imprevistas. [13]

## 4.6 SENSORES WEARABLES

### 4.6.1 Introducción

El término wearable es un concepto novedoso que hace referencia al conjunto de aparatos y dispositivos electrónicos que se incorporan en alguna parte de nuestro cuerpo u objeto, en forma de prenda o complemento. La principal ventaja de estos dispositivos es que no hay que llevarlos en la mano o en el bolsillo, sino que son portátiles y vienen directamente con nosotros. Podemos encontrar muchos ejemplos de dispositivos wearables en nuestro entorno: Smart Watches (relojes inteligentes), Smart Bands (pulseras inteligentes), Smart Glasses (gafas inteligentes), además de su incorporación a prendas de ropa, calzado, etc. Su aplicación abarca muchos sectores, desde el deporte al entretenimiento pasando por el marketing, salud, etc.

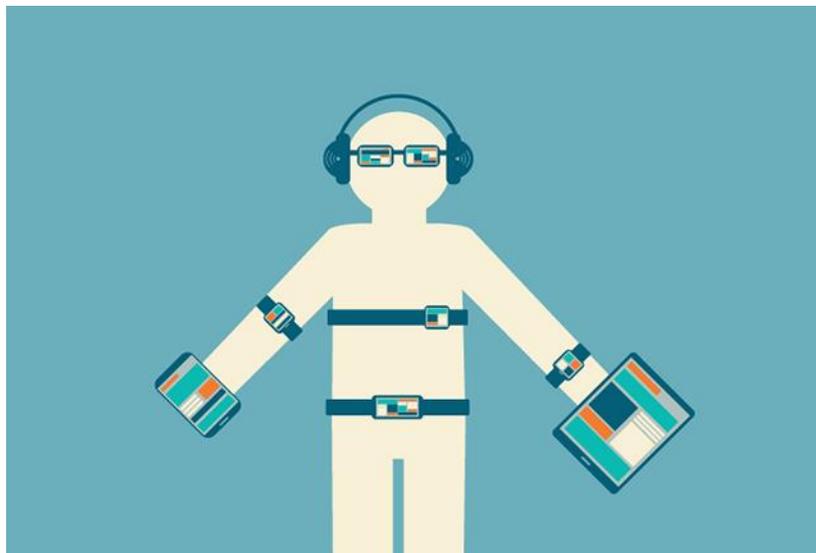
### 4.6.2 Tecnología wearable en la industria

En la Industria 4.0, los wearables pueden tener un papel importante como es en el caso de los habilitadores digitales, enmarcado en el conjunto de tecnologías que hacen posible esta nueva industria que explota el potencial del Internet de las Cosas, convirtiendo elementos físicos en información digital para su posterior tratamiento. Además, pueden beneficiar a los trabajadores en eficiencia, productividad y seguridad aportando valor a los procesos de fabricación.

La utilización de Smart Glasses en plantas de producción permite a los trabajadores recibir información contextualizada de las máquinas que están utilizando. Pueden obtener indicaciones para reparar averías, órdenes de trabajo, etc. todo ello, con las manos libres y sin necesidad de desplazarse de su puesto de trabajo

a consultar dicha información. También pueden incluir cámaras que permiten grabar vídeos o sacar fotografías de forma rápida y, de ese modo, crear un registro documental de las operaciones. La posibilidad de recibir notificaciones, permite que el trabajador esté conectado en todo momento y pueda recibir alertas sobre riesgos potenciales, contribuyendo a la seguridad. También en el área logística las Smart Glasses pueden aportar beneficios, proporcionando información audiovisual sobre los pedidos, soporte sobre la ubicación de los materiales, destinos, etc.

Otro dispositivo que encuentra su sitio en las plantas de fabricación es el Smart Watch. Su utilidad es muy variada, incluyendo opciones como recepción y emisión de alertas, recogida de información biométrica y de posicionamiento GPS o visualización de datos. Se utiliza en líneas de fabricación para que los operarios reciban alertas sobre el funcionamiento de las máquinas y sea posible resolver incidentes sin necesidad de parar la línea de producción. La ventaja de utilizar un Smart Watch sobre otros dispositivos no wearables como un teléfono móvil, es que con una simple vibración el operario es consciente de que hay algún problema y puede recibir información al respecto en el mismo dispositivo. Con un teléfono móvil, puede no enterarse de la vibración y, además, el reloj no interfiere en la forma de trabajar del operario, ya que está en su muñeca y no tiene un gran tamaño. La utilización de Smart Watches también contribuye a la seguridad y prevención de riesgos. Gracias a la incorporación de sensores puede monitorizar movimientos, parámetros biométricos y ambientales y generar una alerta al trabajador y a la central junto con su ubicación.



*Ilustración 31: Sensores wearables*

Estos son algunos ejemplos de utilización de dispositivos wearables en el ámbito industrial, pero las posibilidades de integración de dispositivos con la ropa o el equipamiento de los trabajadores son ilimitadas, así como sus aplicaciones para la

recogida y envío de información y su posterior tratamiento. También es necesario afrontar ciertos retos, como que no sean percibidos por los empleados como un intento de monitorizar o evaluar su trabajo o una forma de vulnerar su privacidad. En cualquier caso, estamos ante un universo de oportunidades y no debemos quedarnos fuera de este viaje emocionante a la nueva industria inteligente y conectada. [14]

## 4.7 SISTEMA SENSORIAL

### 4.7.1 Introducción

En el proyecto se van a emplear tres tipos de sensores, ultrasonidos, de contacto y de fuerza resistivo, y de cada tipo, dos sensores para garantizar una mayor seguridad del robot colaborativo y que el control del mismo sea más eficiente.

### 4.7.2 Capacitive touch sensor switch

Un sensor táctil capacitivo presenta un comportamiento muy similar al de un pulsador, pero a diferencia de éste, el sensor puede activarse con poca o ninguna presión.

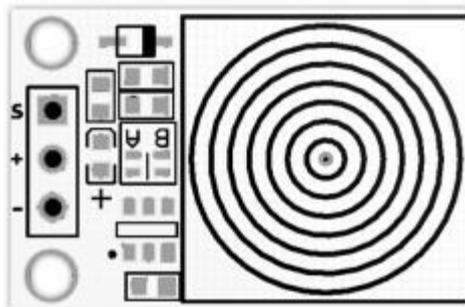


Ilustración 32: Sensor capacitivo VMA 305

Este tipo de sensor hace uso de la variación de la capacidad entre la placa sensora y el cuerpo humano para su funcionamiento.

A medida que la distancia se va reduciendo, la capacitancia aumenta. Gracias a esta acumulación de carga, la placa sensora la detecta y genera una señal digital al superar cierto valor, pudiendo así ser emitida al dispositivo ARDUINO mediante una entrada digital.

La mayor virtud de estos sensores es que no necesita contacto físico para generar la señal, al ser el rango de funcionamiento entre 1-5 mm.

Por el contrario, un inconveniente es que no se pueden emplear en robots de alta velocidad porque el tiempo de respuesta no es lo suficientemente pequeño para que dé tiempo a frenar un robot de gran envergadura.

#### CARACTERISTICAS

- Conexión ..... 3 pines, GND-VCC-Signal
- Voltaje ..... 3.3-5 V
- Dimensiones ..... 30 x 16 x 6 mm
- Peso .....5g

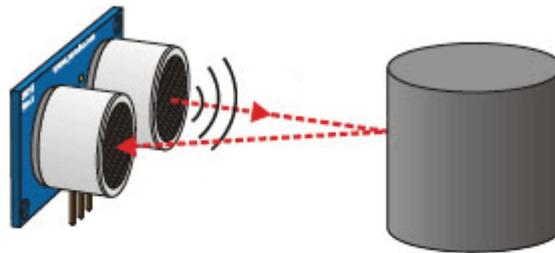
### 4.7.3 HC-SR04 Ultrasonic sensor

Un sensor de ultrasonidos se utiliza para medir distancias. Se envía un pulso de alta frecuencia que no puede captar el ser humano rebotando en los objetos que se encuentren en su camino y posteriormente es reflejado hacia el sensor.



*Ilustración 33: Sensor ultrasónico HC-SR04*

Conociendo la velocidad del sonido y el tiempo entre pulsos, se consigue estimar la distancia con el objeto a determinar.



$$\text{Tiempo} = 2 \cdot (\text{Distancia} / \text{Velocidad})$$
$$\text{Distancia} = \text{Tiempo} \cdot \text{Velocidad} / 2$$

Ilustración 34: Funcionamiento del sensor ultrasónico HC-SR04

Los sensores de ultrasonidos son sensores de baja precisión. La orientación de la superficie a medir puede provocar que la onda se refleje, falseando la medición. Además, no resultan adecuados en entornos con gran número de objetos, dado que el sonido rebota en las superficies generando ecos y falsas mediciones. Tampoco son apropiados para el funcionamiento en el exterior y al aire libre. Como se observa en la siguiente imagen, su rango de funcionamiento eficaz es de unos 30 grados.

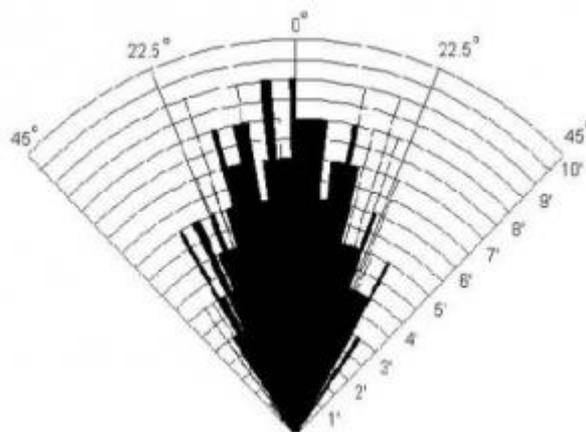


Ilustración 35: Rango de funcionamiento del sensor ultrasónico HC-SR04

Pese a esta baja precisión, que impide conocer con exactitud la distancia a un objeto, los sensores de ultrasonidos son ampliamente empleados. En robótica es habitual montar uno o varios de estos sensores, por ejemplo, para detección de obstáculos, determinar la posición del robot, crear mapas de entorno, o resolver laberintos.

## CARACTERISTICAS

- Voltaje ..... +5 VDC
- Frecuencia ..... 40 KHz
- Resolución ..... 0.3 cm
- Angulo de medición ..... 15°
- Corriente ..... 15 mA
- Señal de entrada Trigger ..... 10  $\mu$ S
- Conexiones ..... 4 pines
- Rango de medición ..... 2-400 cm
- Dimensión ..... 45 x 20 x 15 mm

### 4.7.4 Sensor de fuerza resistivo

El sensor de fuerza resistivo (FSR) es un dispositivo de película de polímero (PTF) que presenta una disminución de la resistencia cuando aumenta la fuerza aplicada a la superficie activa. Su sensibilidad a la fuerza está optimizada para el control por contacto humano de dispositivos electrónicos. Las FSRs no son células de carga o galgas extensiométricas aunque tengan propiedades similares, ni tampoco son adecuadas para medidas de precisión.



*Ilustración 36: Sensor resistivo de fuerza*

Como se ha dicho, la resistencia del FSR cambia a medida que se aplica más presión. Cuando no hay presión, el sensor se parece a una resistencia infinita (circuito abierto). El gráfico que se ofrece a continuación indica aproximadamente la resistencia del sensor en diferentes mediciones de fuerza. (Hay que tener en cuenta que la fuerza no se mide en gramos, se mide en Newtons).

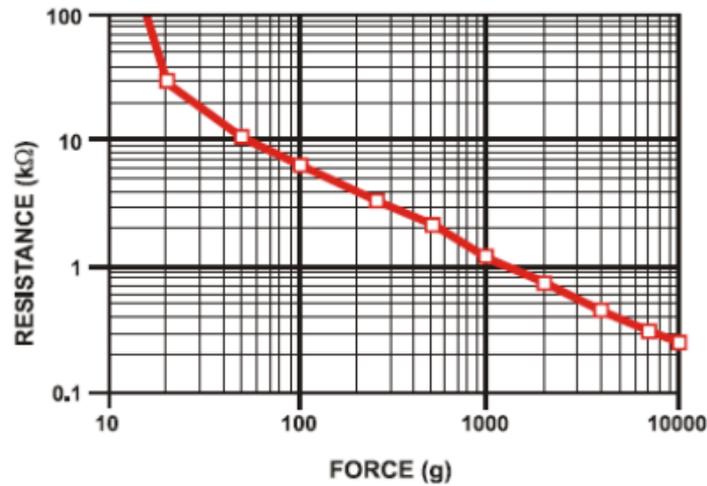


Ilustración 37: Resistencia vs Fuerza

### CARACTERISTICAS

- Rango de fuerza ..... 0.2 N a 20 N
- Resistencia sin actuación ..... > 10 MΩ
- Repetitividad (mismo elemento) ..... ± 2%
- Repetitividad (entre diferentes elementos) ..... ± 6%
- Tiempo de subida ..... < 3 μs
- Histéresis ..... +10% en promedio
- Diámetro total ..... 18.29 mm
- Diámetro área sensible ..... 14.68 mm
- Grosor ..... 0.46 mm
- Recorrido para activarse ..... 0.15 mm
- Vida útil ..... 10 000 000 de activaciones



---

## CAPÍTULO 5

# ***ARDUINO***



Universidad de Valladolid





## 5.1 ARDUINO

Arduino es una plataforma de prototipos electrónica de código abierto (open-source) basada en hardware y software flexibles y fáciles de usar. Es decir, se puede realizar cualquier tipo de proyecto sin necesidad de tener una licencia. Está pensado para artistas, diseñadores, como hobby y para cualquiera interesado en crear objetos o entornos interactivos. Arduino puede sentir el entorno mediante la recepción de entradas desde una variedad de sensores y puede afectar a su alrededor mediante el control de luces, motores y otros artefactos. [15]

## 5.2 HISTORIA

El proyecto Arduino surgió en el año 2005, de mano de unos estudiantes del Instituto de diseño Interactivo IVREA, en Ivrea (Italia), que estaban usando el microcontrolador BASIC Stamp, con un lenguaje de programación simplificado, pero con un coste muy elevado de cada plataforma de desarrollo, alrededor de 100\$. El IVREA estaba a punto de cerrar, y el desarrollo de un sistema Open Hardware no podía ser embargado.

Basándose en el trabajo de un estudiante colombiano Hernando Barragán, quien desarrolló la tarjeta electrónica Wiring, Massimo Banzi, David Cuatrecasas, investigador en el instituto, y Gianluca Martino, un desarrollador local contratado, crearon una plataforma de hardware y software libre, más pequeña y económica, a la que llamaron Arduino.

El curioso nombre del proyecto, tiene su origen en el Bar di Re Arduino (Antiguo Rey Europeo entorno al 1002), donde Massimo Banzi, uno de los fundadores, pasaba parte de su tiempo libre.

Poco tiempo después de terminar el desarrollo, el instituto cerró sus puertas y los desarrolladores intentaron sobrevivir con el nuevo sistema Arduino.

En el 2005 se une al equipo Tom Igoe quien es conocido por sus trabajos en Computación Física (Construcción de sistemas físicos a través de hardware y software que pueden sentir y responder al mundo análogo), quien se encarga de las pruebas del sistema con estudiantes del ITP en Estados Unidos, como también de realizar los contactos para la distribución de la tarjeta en dicho país.

El proyecto gustó mucho, desplazando a otras soluciones del mercado como BasicStamp y los míticos Pics. El mismo Google colaboró en el desarrollo del Kit ADK (Accessory Development Kit), una placa Arduino capaz de comunicarse directamente con teléfonos móviles inteligentes bajo el sistema operativo Android.

Para la producción en serie de la primera versión, se buscó no superar los 30€ de precio y que se ensamblara en una placa azul, que fuese plug and play y compatible con múltiples sistemas operativos: MacOSX, Windows y GNU/Linux. Las primeras 300 unidades se las dieron a los alumnos del Instituto IVREA, con el fin de que las probaran



y empezaran a diseñar sus primeros prototipos. El IDE de Arduino se desarrolló basándose en Processing, buscando la sencillez y la portabilidad a múltiples sistemas operativos.

En la feria Maker Fair de 2011 se presentó la primera placa Arduino 32 bit para trabajar tareas más pesadas, y llegaron nuevas contribuciones al proyecto de parte de Intel, con su placa Galileo.

Tras el enorme éxito del proyecto, aparecieron clones y compatibles y sistemas similares, basados en otros microcontroladores, como Pingüino, basado el PIC 18F. Incluso el propio fabricante de los PIC, microchip, lanzó chipKIT, con PIC32 compatible con el hardware y el software de Arduino. (industriano.es)

Arduino ha seguido creciendo y a día de hoy posee una gran variedad de productos, ya sean placas, accesorios o módulos. [16]

Placas Arduino:

- UNO
- LEONARDO
- 101
- ROBOT
- ESPLORA
- MEGA 2560
- ZERO
- DUE
- MEGA ADK
- PRO
- MO
- MO PRO
- YUN
- ETHERNET
- TIAN
- INDUSTRIAL 101
- LEONARDO ETH
- MKR FOX 1200
- GEMMA
- LILYPAD ARDUINO

## 5.3 ¿POR QUÉ ARDUINO?

Hay muchos otros microcontroladores y plataformas microcontroladoras disponibles para computación física. Parallax Basic Stamp, Netmedia's BX-24, Phidgets, MIT's Handyboard, y muchas otras ofertas de funcionalidad similar. Todas estas herramientas toman los desordenados detalles de la programación de un microcontrolador y la encierran en un paquete fácil de usar. Arduino también simplifica el proceso de trabajo con microcontroladores, pero ofrece algunas ventajas para profesores, estudiantes y aficionados interesados sobre otros sistemas:

**Precio:** Las placas Arduino son relativamente baratas comparadas con otras plataformas microcontroladoras. La versión menos cara del módulo Arduino puede ser ensamblada a mano, e incluso los módulos de Arduino preensamblados cuestan menos de 50\$.

**Multiplataforma:** El software de Arduino se ejecuta en sistemas operativos Windows, Mac OS X y GNU/Linux. La mayoría de los sistemas microcontroladores están limitados a Windows.

Entorno de programación simple y claro: El entorno de programación de Arduino es fácil de usar para principiantes, pero suficientemente flexible para que usuarios avanzados puedan aprovecharlo también. Para profesores, está convenientemente basado en el entorno de programación Processing, de manera que estudiantes aprendiendo a programar en ese entorno estarán familiarizados con el aspecto y la imagen de Arduino.

Código abierto y software extensible: El software Arduino está publicado como herramientas de código abierto, disponible para extensión por programadores experimentados. El lenguaje puede ser expandido mediante librerías C++, y la gente que quiera entender los detalles técnicos pueden hacer el salto desde Arduino a la programación en lenguaje AVR-C en el cual está basado. De forma similar, puedes añadir código AVR-C directamente en tus programas Arduino si quieres.

Código abierto y hardware extensible: El Arduino está basado en microcontroladores ATMEGA8 y ATMEGA168 de Atmel. Los planos para los módulos están publicados bajo licencia Creative Commons, por lo que diseñadores experimentados de circuitos pueden hacer su propia versión del módulo, extendiéndolo y mejorándolo. Incluso usuarios relativamente inexpertos pueden construir la versión de la placa del módulo para entender como funciona y ahorrar dinero. [15]

## 5.4 ARDUINO MEGA 2560

El Arduino Mega 2560 es una placa que se basa en el microcontrolador ATmega2560. Cuenta con 54 pines digitales de entrada/salida (de los cuales 15 se pueden utilizar como salidas PWM), 16 entradas analógicas, 4 UART (puertos serie de hardware), un oscilador de cristal de 16 MHz, un puerto de conexión USB, un conector de alimentación, una cabecera ICSP, Y un botón de reinicio.

El Mega 2560 es una actualización del Arduino Mega al que sustituye.



Ilustración 38: Arduino MEGA 2560



Arduino Mega posee las siguientes especificaciones:

- Microcontrolador: ATmega2560
- Voltaje Operativo: 5V
- Voltaje de Entrada: 7-12V
- Voltaje de Entrada(límites): 6-20V
- Pines digitales de E/S: 54 (de los cuales 15 tienen salida PWM)
- Pines análogos de entrada: 16
- Corriente DC por cada Pin Entrada/Salida: 40 mA
- Corriente DC entregada en el Pin 3.3V: 50 mA
- Memoria Flash: 256 KB (8KB usados por el bootloader)
- SRAM: 8KB
- EEPROM: 4KB
- Velocidad del reloj: 16 MHz
- Largo: 101.52mm
- Ancho: 53.3mm
- Peso: 37g



# ***IMPRESIÓN 3D***



Universidad de Valladolid



## 6.1 INTRODUCCIÓN

Para la fabricación de las piezas que forman una de las dos cintas transportadoras que se van a emplear en el proyecto, se va a utilizar la técnica de la impresión 3D.

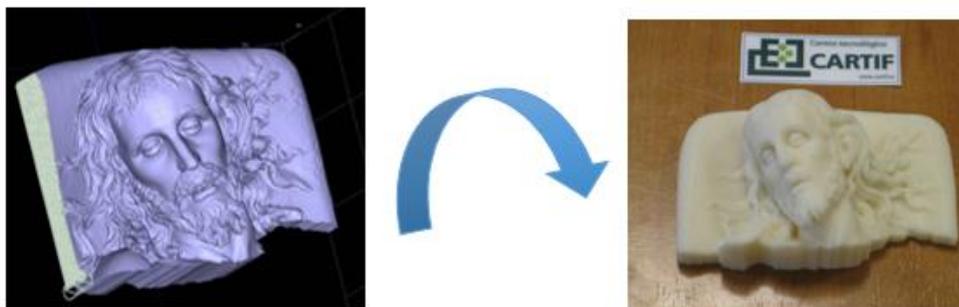
## 6.2 HISTORIA

El inicio de la impresión 3D data de 1976, cuando fue inventada la impresora de inyección de tinta. A partir de ese momento es cuando la tecnología evoluciona para pasar de la impresión con tinta a la impresión con materiales. La impresión 3D ha sufrido cambios durante décadas en diferentes ramas de la industria.

Los sistemas diseño asistido por ordenador (CAD) forman parte del camino en el diseño y desarrollo de nuevos productos.

En estos últimos años se han presentado nuevas formas de producción de modelos y prototipos de productos, que han significado una innovación radical en la forma de entender la producción de prototipos, como la impresión 3D.

El prototipado puede ser definido como el grupo de técnicas que permiten la fabricación, de forma rápida, de un modelo a escala utilizando información suministrada por sistemas CAD. Básicamente consiste en convertir diseños informáticos en objetos reales.



*Ilustración 39: Pieza realizada con impresión 3D*

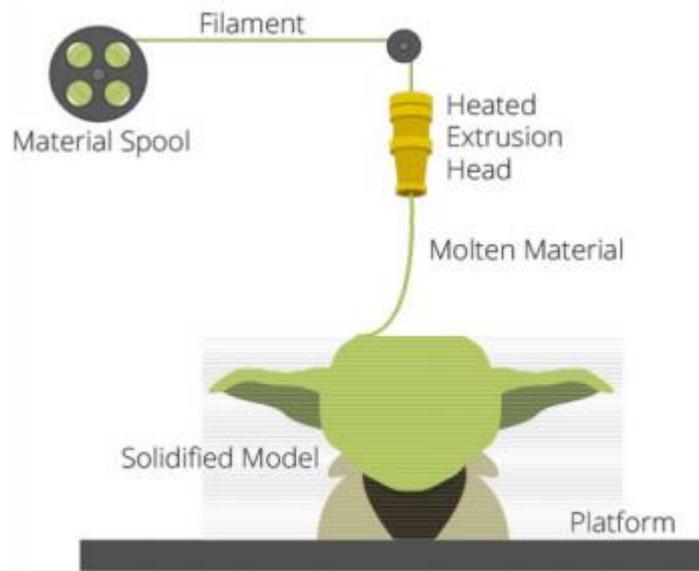
## 6.3 TIPOS DE IMPRESIÓN 3D

Existe un gran número de métodos para la impresión 3D, diferenciándose en la manera de usar las diferentes capas para crear los objetos.

Actualmente, los tres procesos de fabricación aditiva más utilizados son los que se exponen a continuación: [17]

- **MODELADO POR DEPOSICIÓN FUNDIDA (FDM):**

Consiste en depositar polímero fundido sobre una base plana, capa a capa. El material, que inicialmente se encuentra en estado sólido almacenado en rollos, se funde y es expulsado por una boquilla en minúsculos hilos que se van solidificando conforme van tomando la forma de cada capa.



*Ilustración 40: Modelado por deposición fundida*

- **SINTERIZADO SELECTIVO POR LÁSER (SLS):**

Consiste en la colocación de una fina capa de material en polvo en un recipiente a una temperatura ligeramente inferior a la de fusión del material. Un láser sinteriza las áreas seleccionadas causando que las partículas se fusionen y solidifiquen. Se van añadiendo y sintetizando sucesivas capas de material hasta obtener el prototipo deseado. El polvo no solidificado actúa de material de soporte y puede ser reciclado para posteriores trabajos.

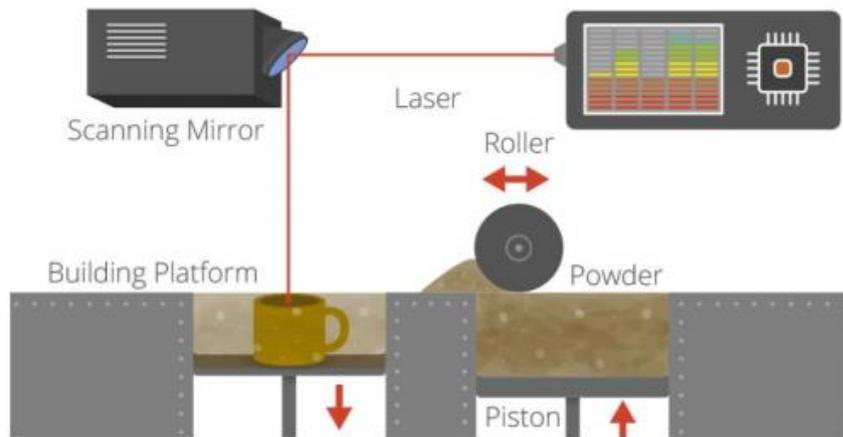


Ilustración 41: Sinterizado selectivo por láser

• **ESTEREOLITOGRAFÍA (SLA):**

Consiste en la aplicación de un haz de luz ultravioleta a una resina líquida (contenida en un cubo) sensible a la luz. La luz ultravioleta va solidificando la resina capa por capa. La base que soporta la estructura se desplaza hacia abajo para que la luz vuelva a ejercer su acción sobre el nuevo baño, así hasta que el objeto alcance la forma deseada.

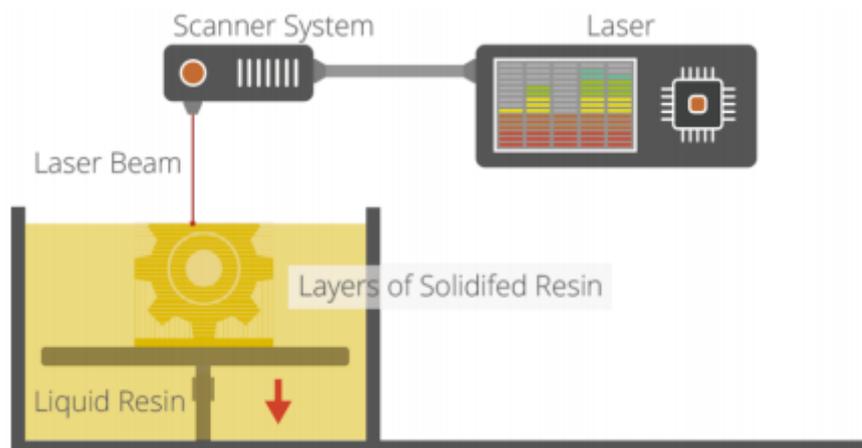
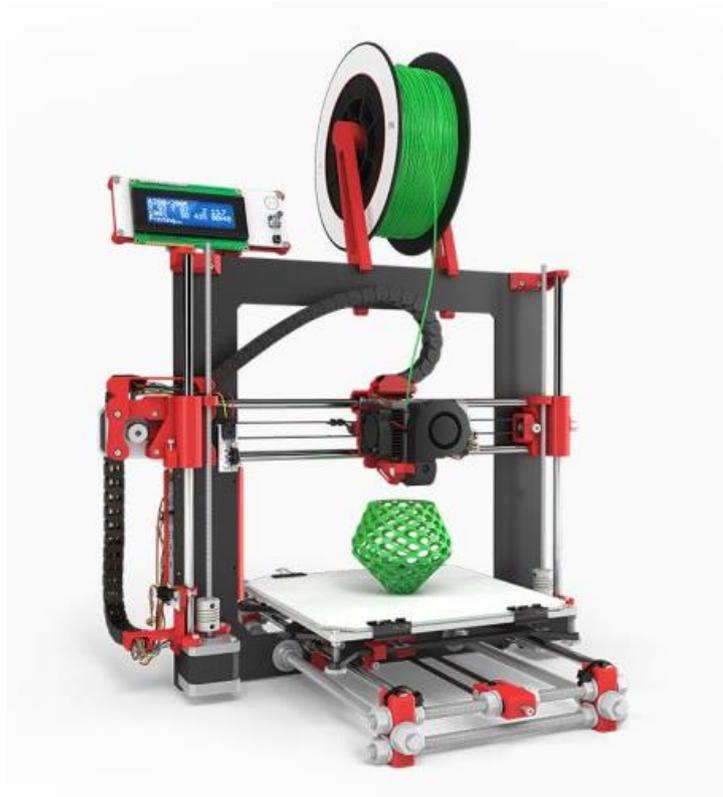


Ilustración 42: Estereolitografía

## 6.4 PRUSA I3

La impresora 3D empleada para la fabricación de las piezas es la Prusa i3 Hephestos.



*Ilustración 43: Impresora Prusa i3*

La Prusa i3 Hephestos está fabricada con la mejor selección de componentes disponibles, algunos de ellos diseño propio de BQ:

- Extrusor de diseño propio
- Pantalla LCD.
- Lector de tarjetas SD.
- Marco de Aluminio.
- Boquilla de 0.4 mm para filamento de 1.75 mm
- Sistema de nivelado



# ***SOFTWARE***



Universidad de Valladolid



## 7.1 SOFTWARE ARDUINO

El software utilizado para la programación de las cintas transportadoras es el software ARDUINO.

El software ARDUINO es un IDE (Entorno de Desarrollo Integrado) con un entorno de programación que ha sido empaquetado como un programa de aplicación; es decir, consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica (GUI). Además incorpora las herramientas para cargar el programa ya compilado en la memoria flash del hardware.

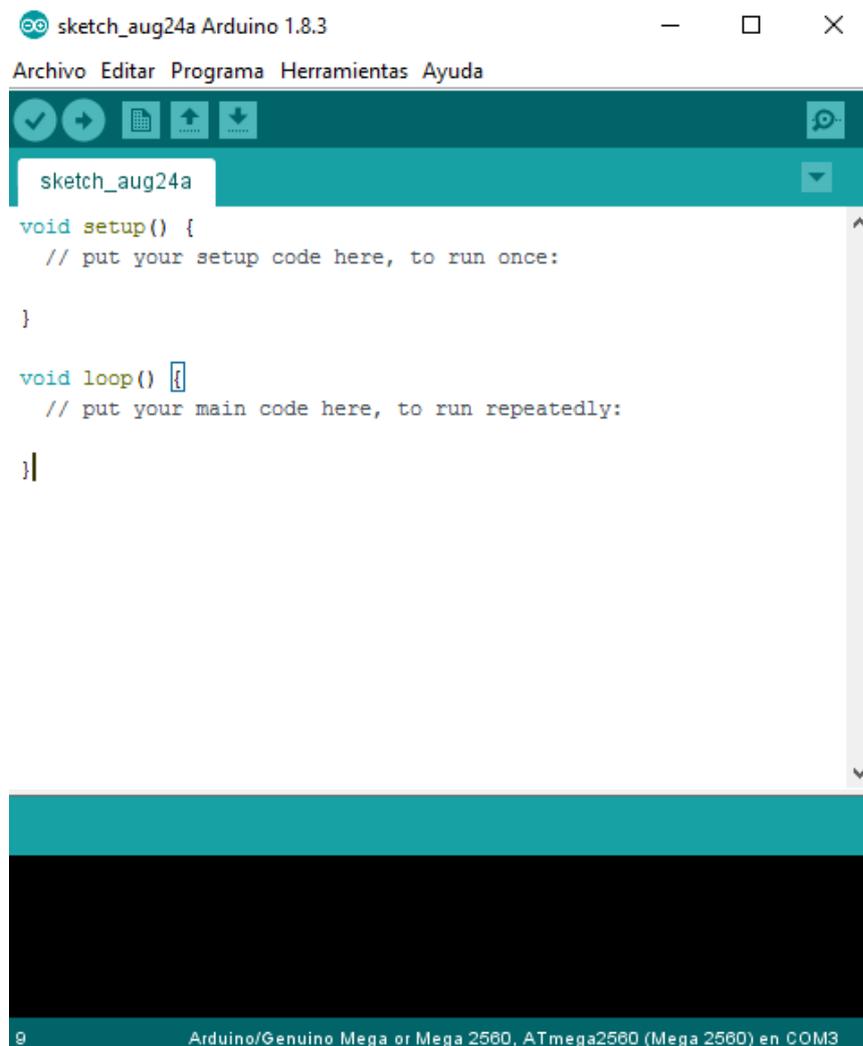


Ilustración 44: Interfaz del software de Arduino

La programación de ARDUINO está basada en el lenguaje C, soportando funciones en C y C++.



Para construir un programa únicamente es necesario definir dos funciones que se ejecutarán de forma cíclica.

- **Void setup ():** Es la función de inicialización del programa, en ella se definen los pines de entrada y salida del sistema. Se ejecuta una sola vez.

- **Void loop ():** Esta función es el cuerpo del programa, se ejecuta en bucle infinito hasta que la placa es desconectada.

Un compilador estándar de C++ podría no reconocer un programa de ARDUINO, por este motivo, cuando pulsamos el botón “Cargar”, se realiza una copia del código en un archivo temporal encabezada por la instrucción #include y la función main(), convirtiéndolo así en un programa perfectamente ejecutable en el lenguaje C++.

### 7.1.1 FUNCIONES BASICAS

- **pinMode (pin, mode):** Función usada en el setup () para configurar el pin especificado y ser ejecutado como entrada o salida (INPUT-OUTPUT). Esta función no tiene ningún valor de retorno.

- **digitalRead (pin):** Lee el valor (HIGH O LOW) desde un pin digital específico y devuelve el valor HIGH (alto) o LOW (bajo). El pin puede ser especificado con una variable o una constante.

- **digitalWrite (pin, value):** Esta función activa (HIGH) o desactiva (LOW) un pin digital.

Si el pin ha sido configurado como OUTPUT, su voltaje pasara a ser de 5v o 3.3 si la tarjeta lo permite.

- **analogRead (pin):** Lee el valor del pin analógico especificado con una resolución de 10 bits, es decir, puede convertir valores entre 0-5 voltios en valores enteros entre 0-1023. Los pines analógicos a diferencia de los digitales no necesitan declararse previamente como entrada o salida.

- **analogWrite (pin, value):** Escribe un valor analógico en el pin indicado usando modulación por ancho de pulso (PWM). Esta función puede servir para iluminar un diodo LED con distintas intensidades o hacer girar un motor a distintos rangos de velocidades. Puede especificarse un valor de 0-255. Un valor 0 genera 0V en el pin especificado y 255 genera 5V.



- **delay (ms):** Realiza una pausa en el programa la cantidad de tiempo en milisegundos especificada en el parámetro. Proporciona una interrupción en el programa el tiempo que se especifique medido en milisegundos.

- **Serial.begin (rate):** Inicializa el puerto serie y establece la velocidad de datos para la transmisión de datos en serie. La velocidad más común para la comunicación con el ordenador es de 9600 bits por segundo.

- **Serial.println (data):** Imprime datos al puerto serie seguido por un retorno de línea automático. Tiene la misma forma que Serial.print () pero este sin el salto de línea al final. Este comando puede emplearse para realizar la depuración de programas. Para ello pueden mandarse mensajes de depuración y valores de variables por el puerto serie y posteriormente activando desde el software Arduino el “Serial Monitor” se puede observar el contenido del puerto serie.

- **Serial.read ():** Lee o captura un byte (un carácter) desde el puerto serie. Devuelve un -1 si no hay ningún carácter en el puerto serie. [18]

## 7.2 CATIA

Para la construcción de las piezas de la cinta transportadora el primer paso es el diseño de las mismas. El software utilizado para ello ha sido CATIA.

Catia es un programa informático de diseño, fabricación e ingeniería, asistida por ordenador de la casa Dassault Systèmes, la misma que otro programa de procesos de ingeniería inversa, Solid Works. El programa proporciona todo tipo de ayudas desde el diseño, hasta la producción y el análisis de los productos a desarrollar, es decir, todo el proceso requerido para diseñar y producir piezas industriales. Provee una arquitectura abierta para el desarrollo de aplicaciones o para personalizar el programa. Las interfaces de programación de aplicaciones se pueden programar en Visual Basic y C++.

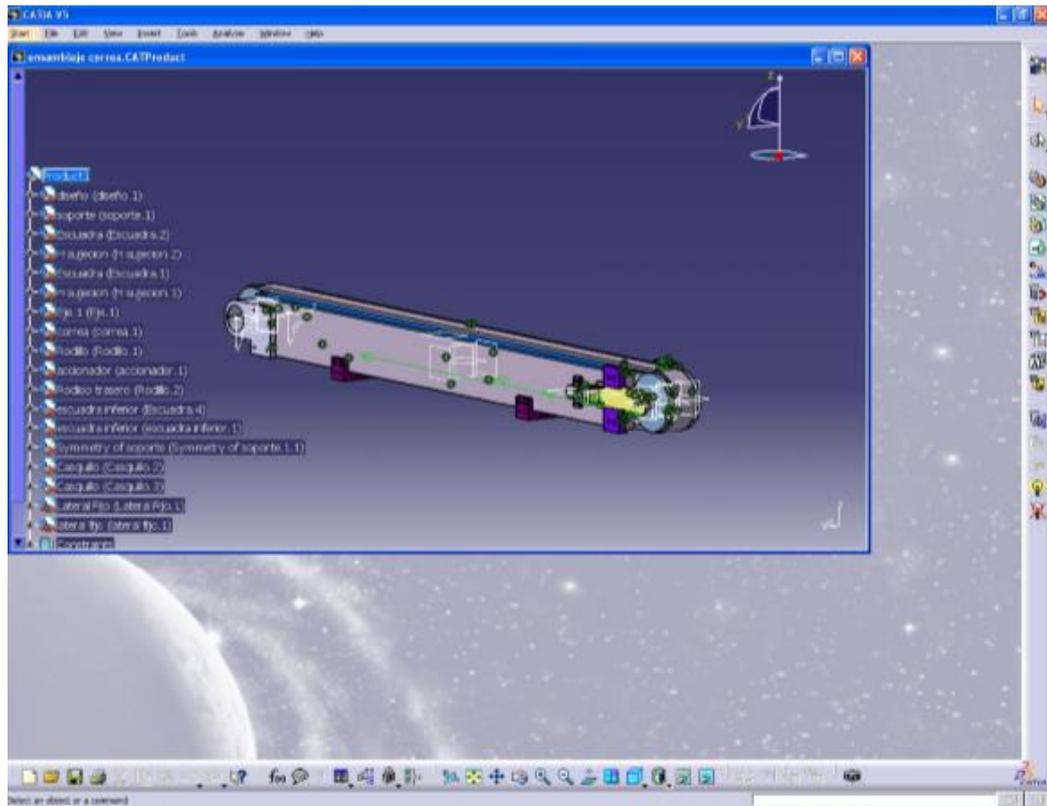


Ilustración 45: Interfaz del software CATIA

## 7.3 CURA

CURA es el software que gestiona la impresora 3D. Es una más de las opciones que se nos ofrece a día de hoy en el laminado basado en Skeinforge pero con opciones de configuración simplificadas y una interfaz gráfica más sencilla e intuitiva que su predecesor. Como ventajas, el software está capacitado para mostrar las piezas en 2D y 3D y el resultado del laminado que previsiblemente se obtendrá una vez finalizada la impresión. Otra de las ventajas de Cura son algunas novedades tales como la posibilidad de mover la aguja por el interior de la pieza cuando no está imprimiendo para evitar hilos erróneos.

Una curiosidad del programa es que tiene como objetivo no sólo actuar como programa de laminado, sino también como anfitrión (host) de la impresión, por lo que intenta englobar todos los procesos necesarios para la impresión de la pieza, es decir, un programa que además de generar el g-code, esté capacitado para enviarlo a la impresora.

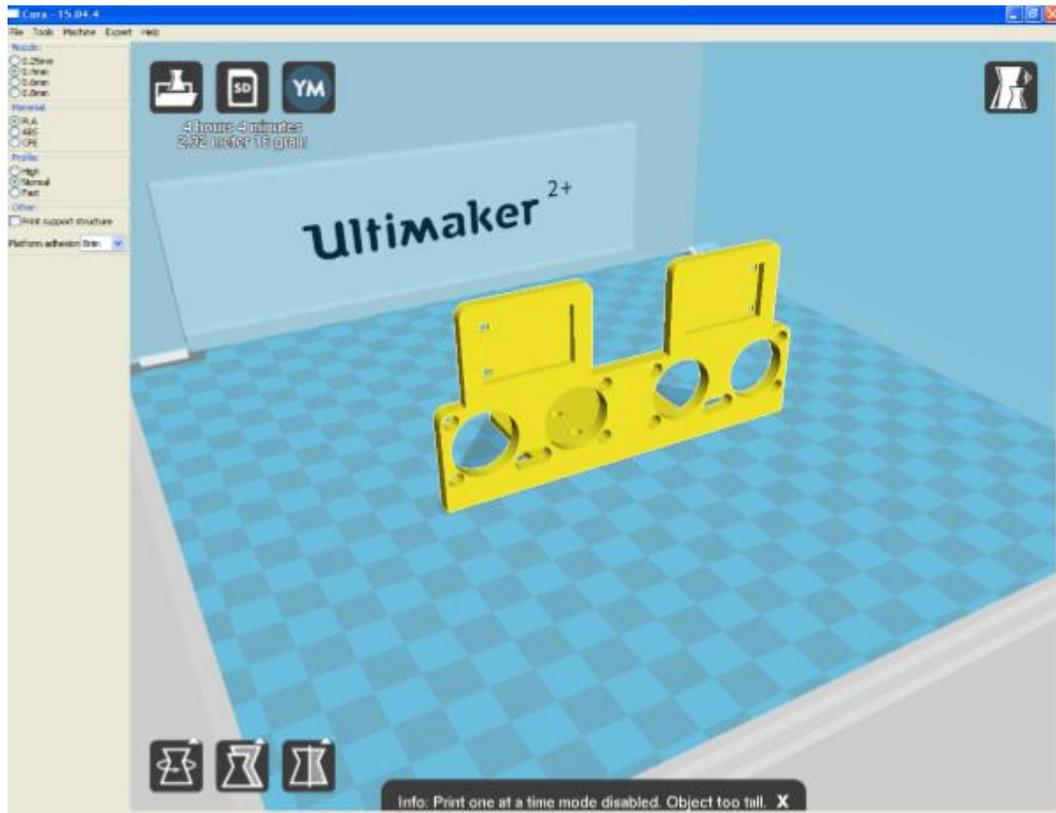


Ilustración 46: Interfaz del software CURA



Universidad de Valladolid





# ***DISEÑO MECÁNICO***



Universidad de Valladolid





## 8.1 INTRODUCCIÓN

La simulación del robot colaborativo se va a llevar a cabo con dos cintas transportadoras que representarán al robot y al operario.

La cinta que hará las funciones de robot ha sido generosamente proporcionada por la empresa de investigación CARTIF y no ha tenido que ser modificada.

Por el contrario, la cinta transportadora que simula al operario o persona ha tenido que ser construida íntegramente. A continuación se detallará paso a paso como ha sido el procedimiento.

## 8.2 DISEÑO

Para comenzar, se han diseñado las diferentes piezas necesarias para la creación de la cinta. El programa utilizado ha sido CATIA.

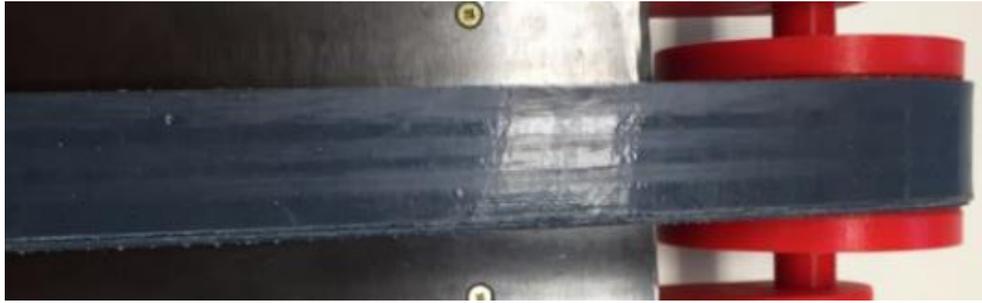
Una vez diseñadas estas piezas, se han generado los archivos correspondientes para que puedan ser utilizados en el programa CURA.

Después de configurar el programa CURA de la manera necesaria para imprimir las piezas de forma óptima, enviamos los archivos .gcode que genera el programa a una tarjeta SD que se insertará en la impresora 3D y se procederá a la impresión de las piezas de la cinta transportadora.

La decisión de realizar la cinta transportadora con piezas de impresión 3D ha sido tomada por la sencillez, el barato coste del material que se utiliza para la impresión 3D y la flexibilidad que nos aporta, dado que si surge algún inconveniente, la capacidad para fabricar otra pieza es mayor y más económica.

### 8.2.1 Banda

La banda que se ha escogido para transportar el soporte que simula al operario que se acerca al robot, es una banda que tenía la empresa CARTIF inutilizada y se acerca a la perfección a las características y tamaño que son necesarios para el proyecto.

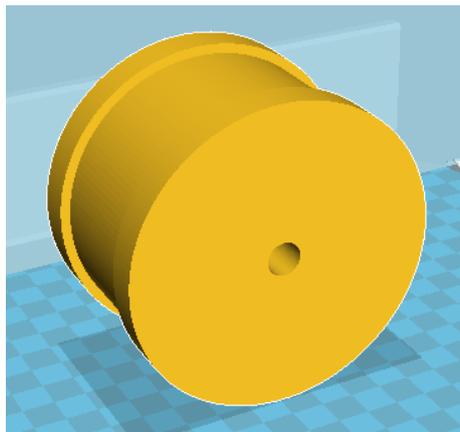


*Ilustración 47: Banda de la cinta transportadora*

## 8.2.2 Rodillo

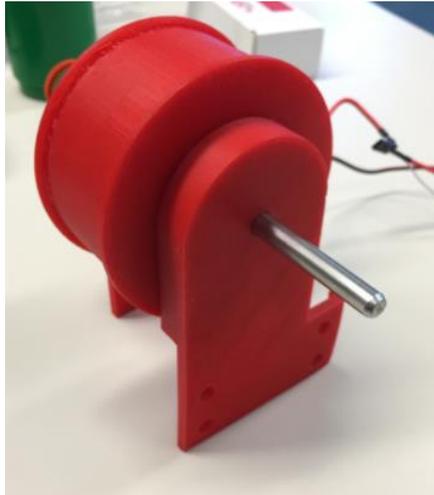
Para cada extremo de la cinta se necesita un rodillo que permita hacer rotar la banda.

Uno para el lado del motor y el otro en el lado opuesto donde se tensa la cinta transportadora. En el lado del motor, hay que ajustarse a éste y tomando las medidas necesarias, el resultado es el que se observa a continuación.



*Ilustración 48: Rodillo en el software CURA*

El rodillo es el mismo para los dos extremos, lo único que varía es el diámetro del eje. En el caso del motor, el taladro que hay que realizar al rodillo ha de ser del mismo diámetro que tiene el eje del motor.



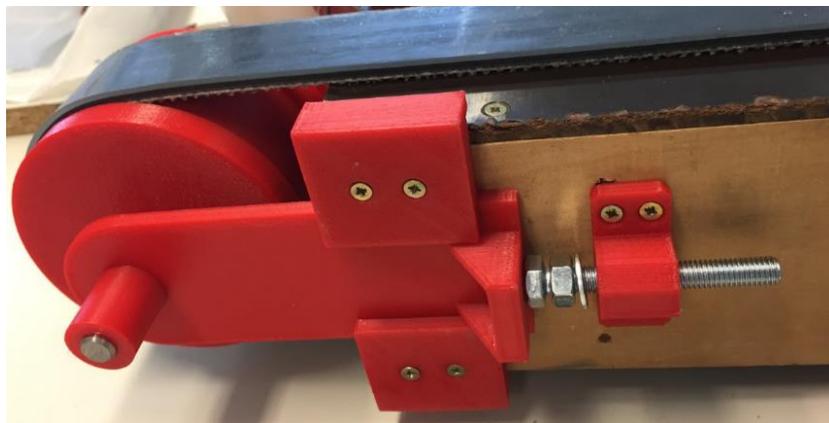
*Ilustración 49: Rodillo después de ser impreso*

En el otro lado de la cinta se ha reutilizado un eje que anteriormente tenía otras utilidades. Los perfiles elegidos para este extremo de la cinta son móviles, para poder colocar el tensor.

### 8.2.3 Tensor

El tensor es una parte fundamental de la cinta transportadora, permite ajustar la tensión de la banda y lograr un contacto adecuado con el rodillo, consiguiendo la transmisión de fuerza entre estas dos partes, evitando el patinaje.

En este caso se han colocado dos tornillos a los lados de la cinta que pueden variar su posición empujando los perfiles móviles del rodillo, ajustándolos en la posición que sea necesaria.

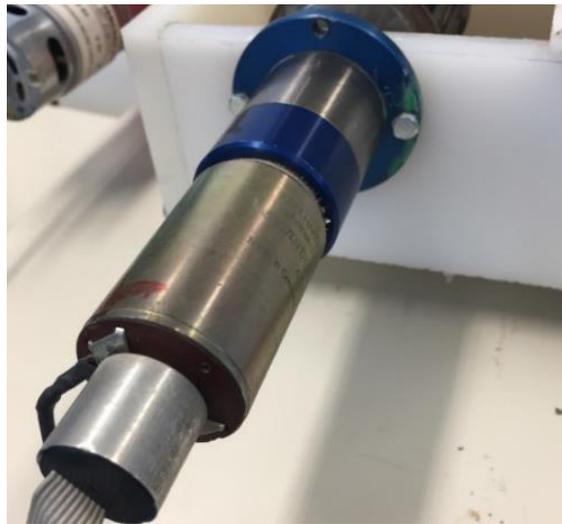


*Ilustración 50: Tensor diseñado para la cinta transportadora*

## 8.2.4 Motores

Cada cinta transportadora va a disponer de un motor de corriente directa que nos permita transmitir el movimiento a los rodillos y con ellos el movimiento de la banda.

La cinta transportadora que ya estaba construida, utiliza un motor de un máximo de 12 voltios de la marca Faulhaber.



*Ilustración 51: Motor Faulhaber de la primera cinta transportadora*

Para la construcción de la segunda cinta, se ha tenido que buscar un motor con los valores necesarios para mover la banda a una velocidad aceptable.

El motor finalmente utilizado ha sido prestado por la empresa CARTIF, el cual no tenía una función actualmente.



*Ilustración 52: Motor Tercesa de la segunda cinta transportadora*

El voltaje máximo que acepta son 24 voltios, un voltaje más alto que el primer motor, lo que va a ser un inconveniente a la hora del control de los mismos. Pero como el precio de un motor es relativamente alto, es preferible reutilizarlos que tener que comprar uno nuevo.

## 8.2.5 Piezas periféricas

Para finalizar el diseño de las dos cintas transportadoras que van a ser el elemento clave del proyecto, faltan por crear las piezas que simularán tanto al robot, como a la persona, operario u objeto que se va a aproximar al robot.

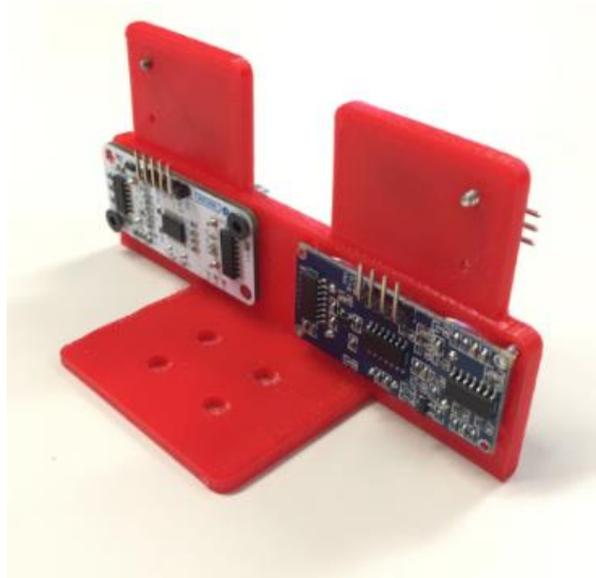
La pieza que va a contener los sensores ha sido diseñada en Catia y posteriormente ha sido impresa con la impresora 3D.

Las medidas han tenido que ser minuciosas para que los sensores se ajusten perfectamente evitando posibles fallos.

Para fijar los sensores ultrasónicos se han utilizado tornillos de métrica 3, sin embargo para los sensores de contacto se dejó una ranura del mismo tamaño de estos para que sean encajados y a mayores se han sujetado con tornillos de un tamaño más pequeño que los de métrica 3, al ser lo que el propio sensor requería.



*Ilustración 53: Parte delantera del soporte de sensores*

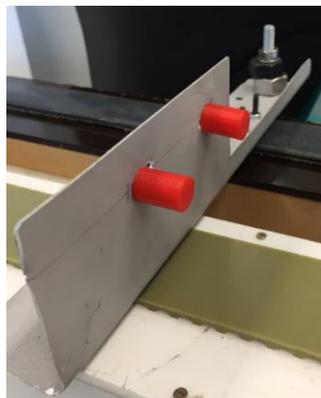


*Ilustración 54: Parte trasera del soporte de sensores*

En el caso del objeto que va a ser sentido, se ha escogido un material ligero como es el aluminio para que la cinta transportadora no tenga problemas cuando se tenga que desplazar.

Ha sido reutilizada una escuadra de aluminio que tenían sin utilizar en la empresa Cartif. Para el diseño de la pieza se ha utilizado la herramienta radial, dando los cortes precisos previamente marcados de la manera que la altura sea la adecuada para que los sensores puedan detectarla.

Como se puede apreciar en la imagen, se han diseñado dos cilindros que están colocados en la altura precisa para que puedan ser detectados por los sensores de contacto. La razón de construir estos dos cilindros es que los sensores de contacto puedan detectar la pieza de aluminio antes de que la placa choque con los sensores ultrasónicos y de esta manera detener los motores y evitar posibles riesgos de accidentes.



*Ilustración 55: Placa sensada y los detectores de contacto*

Para fijar la pieza se han hecho cuatro tornillos a la cinta y otro más grande a la placa de aluminio con el objetivo de hacer de contrapeso, evitando de esta manera que haga contacto y no haya roces con la banda de la otra cinta transportadora.



*Ilustración 56: Contrapeso de la placa de aluminio*

El resultado final de las dos cintas transportadoras se observa a continuación. Para situarlas al mismo nivel, ya que la cinta construida era unos centímetros más alta, se han colocado en la base de la segunda cinta unos tacos de madera del tamaño necesario para que se encuentren a la misma altura las dos cintas.

Por otro lado, para que no estén juntas del todo y haya una pequeña separación se han taladrado dos piezas a la cinta construida, para obtener esa separación que se requiere.

### **8.3 PROBLEMAS SURGIDOS**

En un primer intento por construir la cinta se utilizó un motor D.C. con tres etapas reductoras, el motivo de elegir este motor es por el bajo coste y porque venía con las reductoras incorporadas. Pero el movimiento de la cinta era muy lento y trastabillado donde se alternaban momentos de patinaje de un engrane con el eje, lo que provocaba un aumento de la corriente consumida.



*Ilustración 57: Motor con caja reductora*

También se probó a suprimir una etapa de reducción, generando de esta manera menos par de frenado elevando de esta manera la velocidad de giro del motor. Pero esta variación tampoco surtió efecto en el problema del patinaje. Por último se dejó una única etapa de reducción y los resultados fueron mejores pero sin llegar a aportar un movimiento limpio de la cinta transportadora.

Por este motivo se decide cambiar el motor, sustituyéndole por uno con más potencia, como se ha mencionado anteriormente.

Con este nuevo motor, el movimiento de la banda era más regular y sin producirse patinajes, pero la velocidad de giro es bastante más lenta que la del otro motor.

La solución que permitió una velocidad de giro mayor, fue desmontar el motor y eliminar una de las etapas reductoras.



*Ilustración 58: Reductora del motor Tercesa*

El resultado obtenido fue satisfactorio, elevando la velocidad de giro a unos valores adecuados para el proyecto. Podría eliminarse otra etapa, pero posiblemente



la velocidad sería mayor de la necesaria y podría producir problemas en el movimiento del motor.

---

## CAPÍTULO 9

# ***CONEXIÓN CON ARDUINO***



Universidad de Valladolid



## 9.1 INTRODUCCIÓN

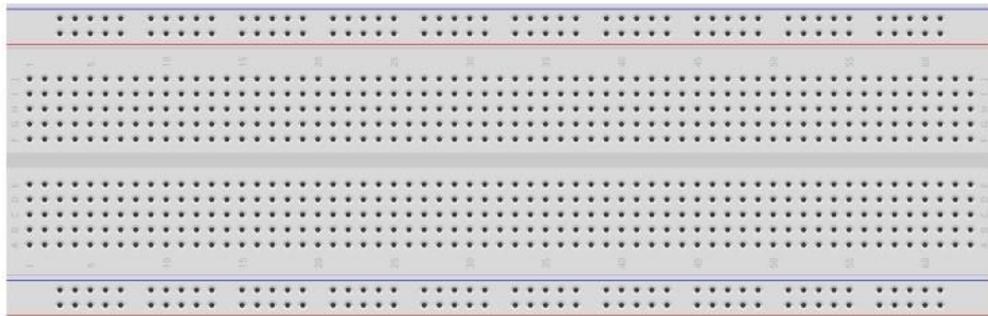
Una vez construidas las cintas transportadoras, el siguiente paso es conectar los motores y los sensores con el cerebro del proyecto, es decir con Arduino, para posteriormente realizar la programación.

Las conexiones se van a dividir en varias etapas para los distintos componentes, para finalmente juntarlo todo y tener el diseño del circuito completo del proyecto.

Para armar estos circuitos va a ser imprescindible el uso de un protoboard, cables de distintos tamaños, y el microcontrolador Arduino.

## 9.2 PROTOBOARD

El protoboard nos sirve de puente entre el Arduino y los distintos componentes. En él se conectarán los cables que unen los motores y sensores con Arduino, y los componentes electrónicos que van a ser necesarios para el funcionamiento del robot colaborativo (diodos, leds, resistencias).



*Ilustración 59: Protoboard*

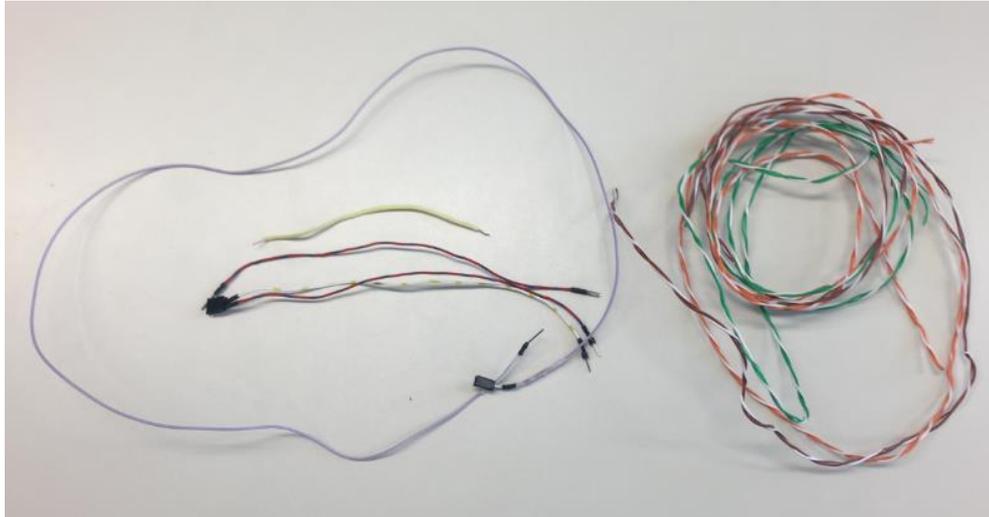
## 9.3 CABLES

Los cables que se van a utilizar van a ser de distintos tamaños en función de la distancia a la que se encuentren los componentes. El protoboard y el Arduino van a estar situados aproximadamente en la mitad de las cintas transportadoras, por lo tanto, los motores y los sensores van a necesitar cables más largos que los que van del protoboard al Arduino al encontrarse a mayor distancia.

Otra distinción que hay que hacer en los cables son los extremos de los mismos. Los cables que salen de los sensores necesitan tener una clavija “hembra” y

el otro extremo tiene que ser “macho” para poder conectarlo al protoboard o al Arduino. El resto de cables van a ser “macho-macho”.

Para fabricar estos cables se han soldado a las puntas distintos pines para facilitar la conexión con los orificios del protoboard o del Arduino.



*Ilustración 60: Cables empleados para la conexión con Arduino*

## 9.4 ARDUINO

Como se viene explicando, el cerebro de este proyecto va a ser el microcontrolador Arduino Mega 2560. En él se van a realizar todas las conexiones necesarias de los sensores, como del protoboard.

A continuación se detallarán y se mostrarán los esquemas de las conexiones de los distintos componentes.

### 9.4.1 Alarma

Se va a simular una alarma con tres diodos led que se encenderán dependiendo de la distancia del sensor ultrasónico con la placa de aluminio.

Si la distancia es lo suficientemente amplia como para que no haya peligro de choque se mantendrá encendido el led verde, si la distancia es considerada intermedia, pasará a encenderse el led amarillo, y en el caso de que haya peligro de contacto será el led rojo el que lucirá.

La conexión es bastante sencilla, los diodos led tienen una patilla más larga que va conectada con una resistencia en serie, esto es debido a que el led se caracteriza por tener una tensión directa y una corriente que circula por él, si la corriente es muy pequeña, el diodo led no se encendería y si la corriente es muy alta, se podría quemar, por lo tanto es necesario una resistencia que evite estos dos

problemas. Estas resistencias van conectadas a los pines digitales de arduino. La patilla más corta del diodo led tiene que conectarse a tierra.

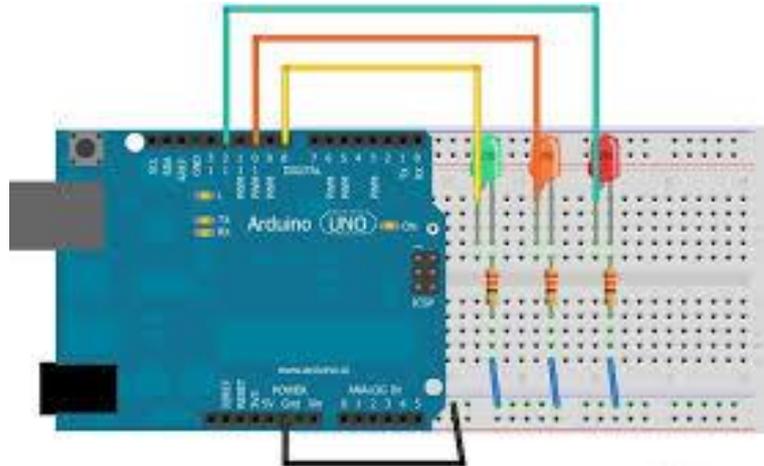


Ilustración 61: Conexiones alarma

## 9.4.2 Ultrasonidos

Los sensores ultrasónicos tienen cuatro patillas y todos se conectan de la misma manera.

Las patillas VCC y GND se conectan a +5 voltios y a tierra respectivamente.

Por otro lado las patillas ECHO Y TRIGGER se conectan a los pines digitales del Arduino.

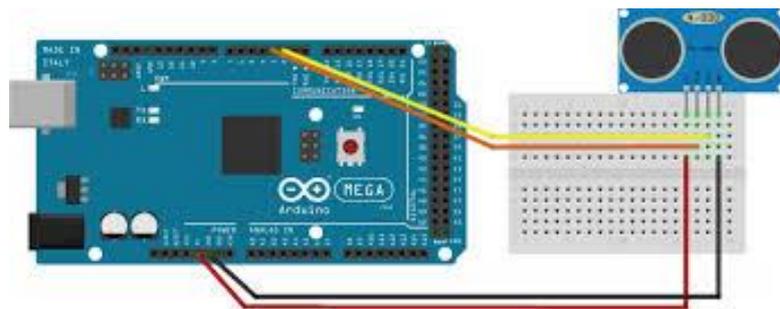


Ilustración 62: Conexiones ultrasonidos

## 9.4.3 Touchless

Las conexiones de los sensores de contacto son todavía más sencillos, constan de tres patillas las cuales se conectan a tierra, a +5 voltios y a un pin digital PWM del Arduino como se puede apreciar a continuación.

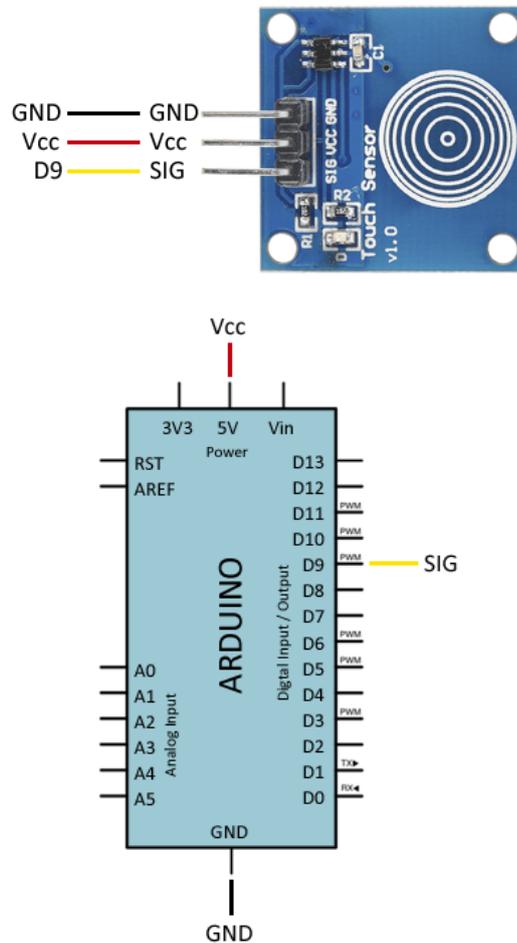


Ilustración 63: Conexiones touchless

### 9.4.4 Sensor resistivo de fuerza

El sensor resistivo de fuerza tiene dos patillas, una va conectada a la tensión de Arduino ya sea 5 voltios o 3,3 voltios y la otra patilla tiene que conectarse a tierra con una resistencia y por otra parte, tiene que conectarse al pin analógico que corresponda de Arduino.

La manera de funcionar, es que a medida que la resistencia del FSR disminuye, la resistencia total del sensor y la resistencia pull-down disminuye. Esto significa que la corriente que pasa a través de ambas resistencias aumenta, lo que a su vez provoca que el voltaje de la resistencia aumenta la resistencia.

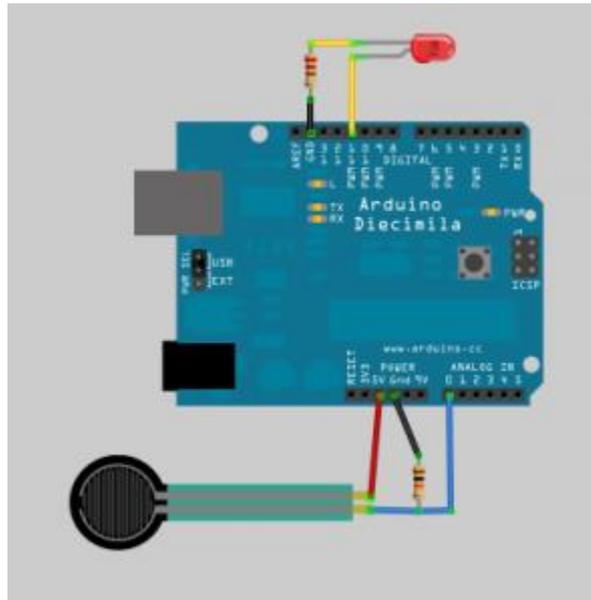


Ilustración 64: Conexión FSR

### 9.4.5 Motores

Por último, tenemos el esquema más complejo de todos y se trata de la conexión de los dos motores.

Primeramente, es preciso la utilización de una fuente de alimentación que va a proporcionar tensión a los dos motores. En este caso Cartif ha prestado una para desarrollar el proyecto.

Para la ejecución de los experimentos es necesario que los motores giren en ambos sentidos. El componente que va a permitir esto es un puente en H (L293B).

Para establecer el giro de un motor de corriente continua se conecta el terminal 1 del motor con el positivo de la alimentación y el terminal 2 con el negativo, si se quiere conseguir la inversión del giro, sería necesario el intercambio de la conexión de los terminales.

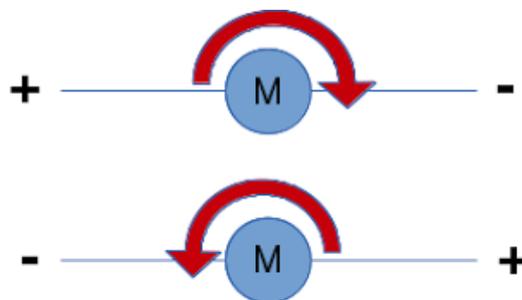


Ilustración 65: Inversión de giro de los motores

Si se quiere conseguir que este cambio sea controlado, se emplean los puentes en H, que consisten en un conjunto de transistores y diodos que nos permite controlar la polaridad de dos terminales de salida en función de unas entradas lógicas. En el proyecto, el puente en H utilizado va a ser el integrado L293B.

Este componente permite controlar dos motores de manera simultánea. los terminales del Motor M1 conectados a los Pines 3 y 6 y los terminales del motor M2 a los terminales 11 y 14, como se ilustra en la figura.

El pin 16 corresponde a la alimentación propia del integrado, 5V, el Pin 8 deberá estar conectado a la tensión que alimentara los motores, puede variar en el rango de (5 a 36) V.

Es importante tener en cuenta que este integrado se alimenta con dos niveles de tensión diferente, uno corresponde a la alimentación propia de integrado, que no debe ser superior a 7V (VSS) y otra es la tensión con la que se alimentarán motores que puede llegar a ser de hasta 36V (VC).

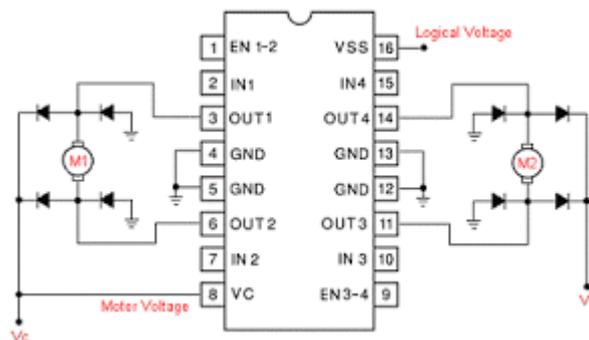


Ilustración 66: Esquema de conexión del integrado L293B

La función que tienen los diodos es la de evitar las corrientes inducidas del motor. En este integrado los diodos no están incluidos como es en el caso del L293D y por lo tanto hay que colocarlos externamente, pudiendo elegir los que mejor se adapten a las necesidades del proyecto, en este caso el diodo escogido es el 1N4007.

Si conectamos el integrado con Arduino y los motores, el diagrama de conexiones es el siguiente.

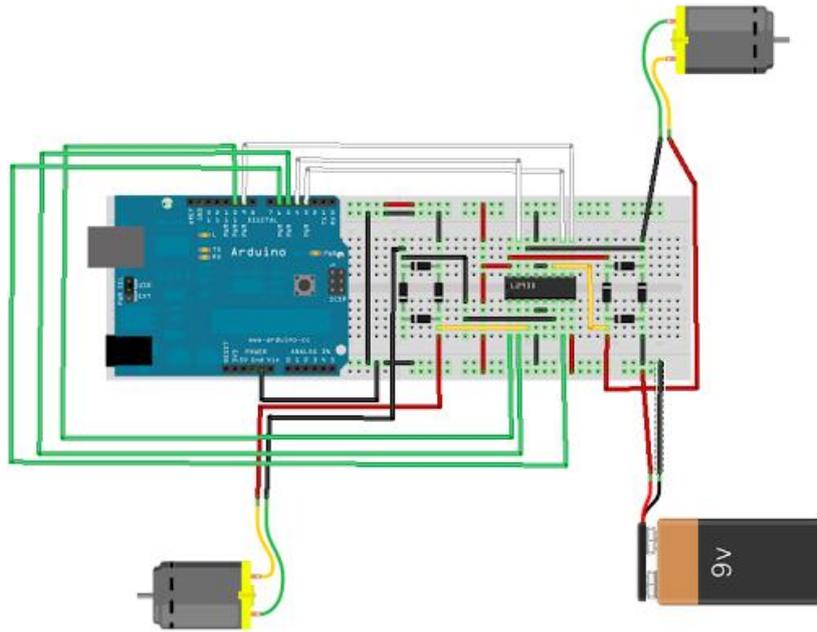


Ilustración 67: Conexión de los motores y el integrado L293B con Arduino

### 9.4.6 Completo

El resultado final de todas las conexiones con los diferentes componentes, es el que se aprecia en la imagen siguiente.

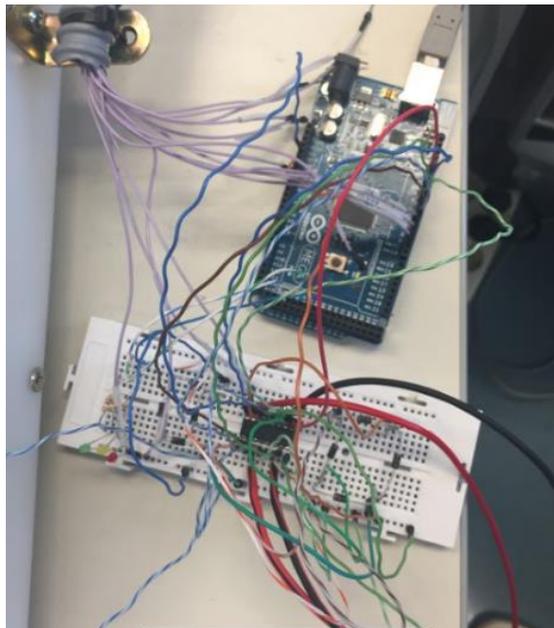
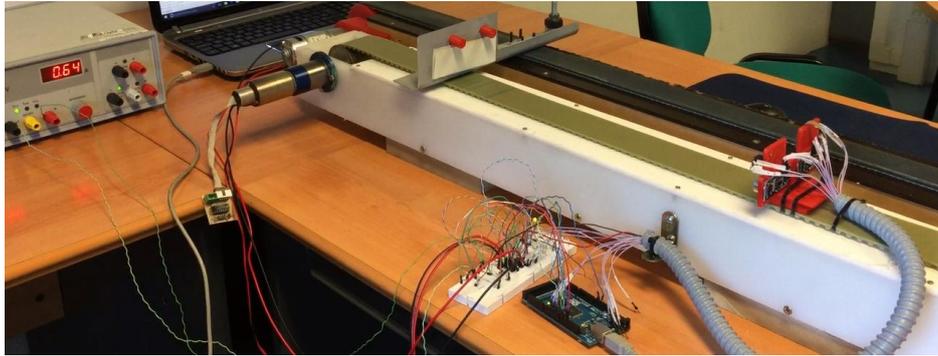


Ilustración 68: Visión de las conexiones entre el protoboard y Arduino



*Ilustración 69: Visión general de las conexiones*



# ***ENSAYO DE EXPERIMENTOS***



Universidad de Valladolid



## 10.1 INTRODUCCIÓN

En el siguiente capítulo se expondrán las pruebas realizadas, los resultados obtenidos y sus conclusiones.

Previamente se ha estudiado en el capítulo 4 los sensores que se van a emplear en el proyecto, así como las conexiones necesarias para el funcionamiento en el capítulo 9.

El primer paso será la calibración de los sensores y comprobar que funcionan de manera correcta, para ello se crean unos sencillos programas en el software de Arduino que hagan funcionar los sensores por separado.

## 10.2 CALIBRACION DE LOS SENSORES

En primer lugar se prueban los **sensores ultrasónicos** con el siguiente programa:

```
const int EchoPin2 = 4;
const int TriggerPin2 = 5;

void setup() {
  Serial.begin(9600);
  pinMode(TriggerPin2, OUTPUT);
  pinMode(EchoPin2, INPUT);
}
void loop() {
  int cm = ping(TriggerPin2, EchoPin2);
  Serial.print("Distancia: ");
  Serial.println(cm);
  delay(1000);
}
int ping(int TriggerPin, int EchoPin) {
  long duration, distanceCm;

  digitalWrite(TriggerPin, LOW); //para generar un pulso limpio ponemos a LOW 4us
  delayMicroseconds(4);
  digitalWrite(TriggerPin, HIGH); //generamos Trigger (disparo) de 10us
  delayMicroseconds(10);
  digitalWrite(TriggerPin, LOW);

  duration = pulseIn(EchoPin, HIGH); //medimos el tiempo entre pulsos, en microsegundos
  distanceCm = duration * 10 / 292 / 2; //convertimos a distancia, en cm
  return distanceCm;
}
```

*Ilustración 70: Programa sensor ultrasónico*

El programa genera un pulso eléctrico en el pin “Trigger” que es el que genera el disparo que durará 10us. Con la función pulseIn se mide el tiempo entre pulsos, que a través de la fórmula que se observa en la imagen se convierte el tiempo en la distancia en centímetros que se quiere medir.

Una vez compilado el programa, se comprueba como la medida que proporciona el sensor es prácticamente la misma que se obtiene midiendo con un metro. Por lo tanto se puede decir que los sensores ultrasónicos funcionan correctamente.

Los segundos sensores que se van a comprobar son los **sensores capacitivos touchless**.

En primer lugar se comprueban los sensores con una entrada digital que nos dé un estado “HIGH” si el sensor percibe un contacto y un estado “LOW” si no hay contacto alguno.

```
const int sensorPin = 8;

void setup()
{
  Serial.begin(9600);
  pinMode(sensorPin, INPUT);
}

void loop()
{
  int estado = digitalRead(sensorPin);

  //Se manda un mensaje al puerto serie en función del valor leído
  if (estado == HIGH)
  {
    Serial.println("Contacto detectado");
  }else
  {
    Serial.println("Contacto no detectado");
  }
  delay(1000);
}
```

*Ilustración 71: Programa sensor de contacto digital*

Los sensores no consiguen detectar contacto alguno y se procede a programar de nuevo pero en este caso con una entrada analógica para visualizar los valores que aportan los sensores y a partir de ahí tomar conclusiones. El programa es el siguiente:

```
void setup()
{
  Serial.begin(9600);
}

void loop()
{
  int estado = analogRead(A0);

  Serial.println(estado);
  delay(1000);
  /* Nos muestra por pantalla el valor entre 0 y 1023
  que proporciona el sensor*/
}
```

*Ilustración 72: Programa sensor de contacto analógico*

Es un sencillo programa que lo único que hace es devolver el valor PWM que tiene el sensor cada segundo.

Los valores que se observan en el monitor serie cuando no hay contacto oscilan entre el número 800 y 900 sin llegar a estabilizarse. Una vez que se presiona con el dedo, se ve que el valor varía los primeros 5 segundos ascendiendo al número 1000, pero a partir de ahí si se vuelve a producir un contacto, el sensor ya no lo lee y por lo tanto es como si no establece el contacto.

La calibración y el funcionamiento de estos sensores capacitivos no es precisa ni correcta, y finalmente no se podrá actuar con ellos en el proyecto debido a que cuando se establezca el contacto entre las dos placas que simulan al robot colaborativo y al operario, la acción de detenerse los motores tiene que ser inmediata, sin correr riesgos de que no lleguen a detectar el contacto los sensores.

Por último, solo falta la calibración del **sensor resistivo de fuerza**, la programación correspondiente es la que se observa en la siguiente ilustración:



```
int fsrAnalogPin = 0;
int LEDpin = 11;
int fsrReading;
int LEDbrightness;

void setup() {
  Serial.begin(9600);
  pinMode(LEDpin, OUTPUT);
}

void loop() {
  fsrReading = analogRead(fsrAnalogPin);
  Serial.print("Analog reading = ");
  Serial.println(fsrReading);

  delay(100);
}
```

*Ilustración 73: Programa sensor resistivo de fuerza*

El programa es similar al del sensor de contacto y la finalidad es la misma, comprobar los valores que proporciona el sensor cuando hay y cuando no hay contacto.

Como es de esperar, el valor que muestra el monitor cuando no hay contacto alguno es 0, y a medida que entra en contacto el dedo de una persona o un objeto, cuanto más fuerza implique sobre el sensor, el valor se irá incrementando llegando como máximo a 1023.

Por lo tanto, se utilizará este tipo de sensor en lugar del sensor capacitivo al ser fiable y preciso para la aplicación del proyecto.

## 10.3 PRUEBA DE EXPERIMENTOS

Una vez calibrados los sensores y sabiendo cuales se pueden utilizar correctamente y cuales no, se procede a realizar los distintos experimentos.

Se van a llevar a cabo dos distintos tipos de experimentos que corresponderán a dos de las cuatro operaciones colaborativas que nos proporciona la Norma ISO/TS 15066:2016. Estas dos operaciones son: Control de velocidad y separación y limitación de fuerza y energía.

### 10.3.1 Control de velocidad y separación

El primer experimento que se va a llevar a cabo es una prueba de control de la velocidad y la distancia. Para ello únicamente van a ser necesarios los sensores de ultrasónicos. La utilización de dos sensores es debida a una mayor seguridad en el control del proyecto.

Esta prueba consiste en ir disminuyendo la velocidad de los motores dependiendo de la distancia que hay entre los sensores y la placa que simula al trabajador, hasta el punto que la distancia sea mínima (aproximadamente 4 centímetros), cuando se pararán por completo los motores evitando el contacto.

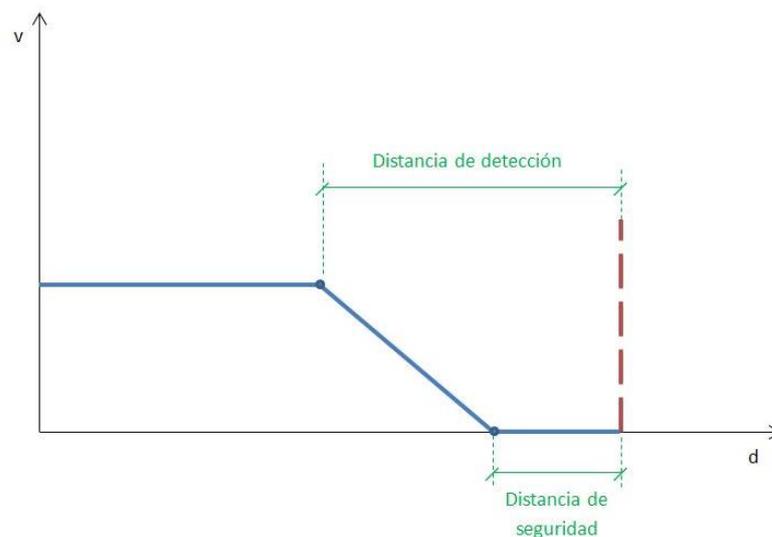


Ilustración 74: Control de velocidad y separación

Se parte de una posición inicial de aproximadamente unos 43 centímetros entre los sensores y la placa y a las distancias de 30 y 12 centímetros, la velocidad de los motores se ven reducidas llegando a detenerse cuando la distancia es de 4 centímetros. Un sistema de alarmas diseñado con tres diodos LED (verde, naranja y rojo) irán indicando respectivamente la “peligrosidad” de posible choque, luciendo el LED rojo cuando la distancia es la mínima (4 cm).

Una vez que los motores se han parado, se produce un delay de 1,5 segundos y los motores invertirán sus sentidos de giro y regresarán a la posición inicial, variando igualmente las velocidades a las mismas distancias que lo realizan cuando las placas se van acercando. Finalmente cuando la distancia sea de 43 cm, los motores se detendrán y esperarán tres segundos para repetir de nuevo todo el proceso.

Para que se produzca la inversión de los motores se ha inicializado una variable “estado” que cambiará en el momento que los motores se detienen.

### 10.3.2 Limitación de fuerza y energía

La diferencia de esta prueba con la anterior es la utilización del sensor de fuerza resistivo.

Los dos motores irán girando de forma que las placas se vayan acercando a una velocidad constante, hasta que el sensor resistivo perciba el contacto de los detectores colocados en la placa que simula al operario. El valor a partir del cual el contacto será percibido es 10, para el no haya problemas en detectar las piezas ni tampoco que se pueda producir un problema con el choque de las placas.

En este momento los motores se detendrán, esperaran un segundo y realizaran la maniobra de invertir el giro para regresar a la posición de partida que es la misma que en el apartado anterior.

Para este caso el sistema de alarma contara con un único diodo LED (rojo), que lucirá cuando se produzca el contacto.

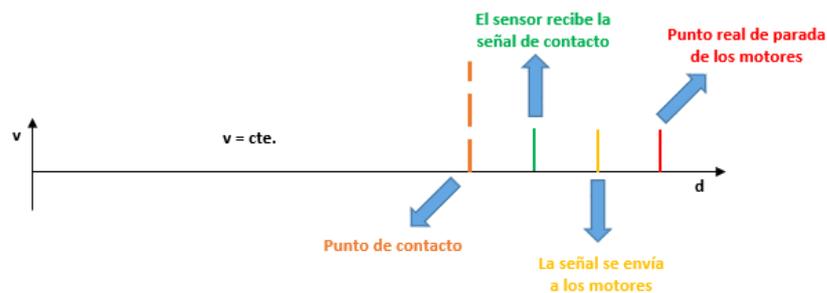


Ilustración 75: Limitación de fuerza y energía

Como se observa en la imagen una vez que se produce el contacto entre el detector y el sensor existen varios retardos, hay un tiempo mínimo entre que se produce el contacto y el sensor recibe la señal de contacto, también hay un delay hasta que se envía la señal a los motores de que tienen que detenerse, y otro retardo hasta que se detienen por completo.

Esto no es un peligro para el proyecto porque las velocidades y las fuerzas que se manejan son pequeñas, el problema surgiría en un robot convencional industrial donde las velocidades y los peligros son mucho mayores. Para evitar esto, se tendrían que utilizar sensores muchos más precisos con unos tiempos de respuesta prácticamente nulos.



## 10.4 CONCLUSIONES

Los resultados obtenidos en los experimentos son los realmente deseados. La simulación de un robot colaborativo se muestra a la perfección cumpliendo la Norma ISO y proporcionando la seguridad necesaria a los trabajadores.

Se podrían realizar más pruebas variando las velocidades de los motores y las distancias a las cuales se reduce la velocidad y se detienen los mismos.



Universidad de Valladolid





# ***ESTUDIO ECONÓMICO***



Universidad de Valladolid



## 11.1 INTRODUCCIÓN

En el siguiente capítulo se va a realizar el estudio económico que supone la realización del proyecto. Para ellos se dividirá en dos apartados; costes directos y costes indirectos para finalmente ponerles en común y calcular el coste total del proyecto.

## 11.2 COSTES DIRECTOS

Los costes directos incluyen los costes asociados al personal y tiempo empleado para el proyecto, hardware y software y los materiales directos utilizados.

### 12.2.1 Costes del personal

El proyecto ha sido realizado por un ingeniero que se ha encargado del desarrollo mecánico y de control del sistema.

Para calcular estos costes, primero tenemos que conocer el coste por hora de un ingeniero y después multiplicarlo por las horas trabajadas.

Se supondrá un salario anual del ingeniero de: **30.000 €**

A continuación se calculan los días laborables en un año:

DÍAS LABORABLES	
Año	365 días
Sábados y Domingos	-105 días
Días de vacaciones efectivos	-20 días
Días festivos reconocidos	-15 días
Días perdidos estimados	-5 días
<b>TOTAL DE DÍAS LABORABLES</b>	<b>220 días</b>

Tabla 1: Días laborables en un año

Una vez conocido el número de días laborables y sabiendo que la jornada laboral es de 8 horas, se obtienen las horas que se trabajan al año:

$$220 \text{ días/año} \times 8 \text{ horas/día} = \mathbf{1.760 \text{ horas/año}}$$

Conociendo el salario anual de un ingeniero y las horas totales trabajadas a lo largo de un año, se obtiene el coste por hora del trabajador:

$$\frac{\text{Coste}}{\text{hora}} = \frac{30.000 \frac{\text{€}}{\text{año}}}{1.760 \frac{\text{horas}}{\text{año}}} = 17,04 \text{ €/hora}$$

El desglose de la distribución de horas empleadas en el proyecto se muestra en la siguiente tabla:

DISTRIBUCIÓN TEMPORAL DEL PROYECTO	
Formación y documentación	100 horas
Estudio del problema	100 horas
Desarrollo mecánico	150 horas
Programación	100 horas
Elaboración de la documentación	200 horas
<b>TOTAL HORAS DEL PROYECTO</b>	<b>650 horas</b>

Tabla 2: Horas dedicadas al proyecto

El coste directo del personal se halla multiplicando el coste por hora del ingeniero y el total de horas empleadas en el proyecto:

$$650 \text{ horas} \times 17,04 \text{ €/hora} = 11.076 \text{ €}$$

### 12.2.2 Costes de amortización de hardware y software

En este apartado cabe señalar que primero se calcularan los precios totales de los productos para después calcular el valor amortizado el tiempo en el que han sido utilizados los distintos elementos.

El tiempo de amortización estimado para un producto informático es de 5 años, que es el tiempo de vida útil que tienen estos aparatos aproximadamente. Por lo tanto hay que calcular los costes con esta apreciación.

Para hallar la amortización durante un año, el coeficiente por el que hay que multiplicar los importes totales es 0,2.

MATERIAL	IMPORTE (aprox.)	AMORTIZACIÓN
Sistema operativo Windows 10	250 €	50 €
Paquete ofimático Microsoft Office 2013	150 €	30 €
Software CATIA	1000 €	200 €
Software CURA	0 €	0 €
Software Arduino	0 €	0 €
Ordenador HP Pavilion dv6	900 €	180 €
Arduino MEGA 2560	50 €	10 €
Sensores (ultrasonidos, contacto y fuerza)	25 €	5 €
<b>TOTAL MATERIAL DE INFORMÁTICA</b>	<b>2375 €</b>	<b>475 €</b>

Tabla 3: Costes amortización del material de informática

Por lo tanto el coste total por hora de los elementos de hardware y software se calculan con la división del coste de la amortización en un año entre el número de horas de uso al año.

$$\frac{\text{Coste final}}{\text{hora}} = \frac{475 \frac{\text{€}}{\text{año}}}{1.760 \frac{\text{horas}}{\text{año}}} \approx 0,27 \text{ €/hora}$$

Se ha empleado como tiempo de uso el tiempo total horas trabajadas al año debido a que en todas las fases del proyecto van a ser necesarios los materiales informáticos.

Por lo tanto ya solo quedaría ajustarlo a nuestro proyecto multiplicando por las horas totales del proyecto.

$$650 \text{ horas} \times 0,27 \text{ €/hora} = 175 \text{ €}$$

### 12.2.3 Costes de material de oficina

Otro tipo de coste directo a tener en cuenta es el coste de papelería, es decir, todo el material de oficina empleado para el desarrollo del proyecto. Este material es necesario para la realización de las diferentes fases del trabajo: desarrollo, edición, impresión de listados, manuales e informes, almacenamiento de programas y documentos, etc.

MATERIAL	IMPORTE (aprox.)
Material de papelería	20 €
Impresiones de documentación	150 €
<b>TOTAL MATERIAL DE OFICINA</b>	<b>170 €</b>

Tabla 4: Costes de material de oficina

### 12.2.4 Costes directos totales

Una vez obtenido todos los costes directos por separado, se procede a la suma de todos ellos para hallar los costes directos totales del proyecto.

MATERIAL	IMPORTE
Personal	11.076 €
material informática	175 €
material de oficina	170€
<b>TOTAL COSTES DIRECTOS</b>	<b>11.421 €</b>

Tabla 5: Total costes directos

<b>COSTES DIRECTOS</b>	<b>11.421 €</b>
------------------------	-----------------

## 11.3 COSTES INDIRECTOS

Los costes indirectos son los gastos producidos por la actividad requerida para la elaboración del proyecto y que no se pueden incluir en ninguno de los gastos directos.

COSTES INDIRECTOS	
Dirección y servicios administrativos	150 €
Costes de electricidad	150 €
Costes de telefonía	50 €
Costes de desplazamiento	400 €
<b>TOTAL COSTES INDIRECTOS</b>	<b>750 €</b>

Tabla 6: Total costes indirectos

<b>COSTES INDIRECTOS</b>	<b>750,00 €</b>
--------------------------	-----------------

## 11.4 COSTES TOTALES

Los costes totales son el resultado de sumar los gastos directos e indirectos, siendo el montante total para este proyecto:

COSTES TOTALES	
Costes directos	11421 €
Costes indirectos	750 €
<b>COSTE TOTAL DEL PROYECTO</b>	<b>12.171 €</b>

Tabla 7: Coste total del proyecto



En conclusión, el coste total del proyecto asciende a la cantidad de:

**COSTES TOTALES DEL PROYECTO**

**12.171 €**



# ***CONCLUSIONES Y LÍNEAS FUTURAS***



Universidad de Valladolid





## 12.1 CONCLUSIONES

En este apartado se expondrán las conclusiones del trabajo en referencia a los objetivos marcados al principio del mismo.

En primer lugar, la investigación sobre robótica colaborativa ha sido muy productiva. Introducirse en el mundo de la robótica hace ver el futuro de una manera distinta.

La necesidad de ser competitivo frente a las empresas rivales, tiene como consecuencia en las empresas tener que evolucionar rápidamente, y un claro ejemplo de ello es la robótica colaborativa.

Este proyecto ha hecho conocer el microcontrolador Arduino, un componente electrónico muy sencillo de utilizar por su contenido open-source, que te permite crear, programar y modificar una gran cantidad de proyectos de distintos tipos sin ningún problema. Gracias a esta facilidad de programación, se puede llegar a más personas para que se introduzcan en el mundo de la electrónica y la automática.

A su vez, otro término del proyecto, característico de la industria 4.0, es la impresión 3D. Un concepto que cada vez va cogiendo más fuerza y que también evoluciona a una velocidad muy elevada.

Gracias a esta técnica se ha podido construir este proyecto de una manera económica y eficaz.

Por último, las conclusiones del desarrollo de los experimentos son satisfactorias, con el único inconveniente de los sensores utilizados. El tiempo de respuesta de éstos y la fiabilidad no es su mayor fuerte, llevando a complicarse la programación y la ejecución de las pruebas.

Los sensores ultrasónicos por su parte han funcionado bien y las distancias que medían estaban acorde con la distancia real. El tiempo que se tarda desde que se lee la distancia del sensor hasta que el motor cambia la velocidad, es el suficiente para poder trabajar sin problemas.

Por el contrario, los problemas se han dado en los sensores de contacto.

En primer lugar, para calibrarlos, los valores que marcaban en el Monitor Serial de arduino permanecían constante en los primeros instantes, pero a medida que pasaba el tiempo, estos valores iban variando haciendo muy complicado su calibración.

Además de esto, una vez detectado el objeto, los valores variaban correctamente en los primeros instantes, pero cuando el tiempo pasaba, de nuevo el sensor dejaba de detectar el objeto, lo que resultaba prácticamente imposible poder trabajar con estos sensores de contacto.

Para solucionar estos problemas, fueron sustituidos los sensores de contacto por un sensor resistivo de fuerza, el cual proporcionó una fiabilidad y un tiempo de respuesta adecuados para el proyecto.



El resultado final que se quería conseguir era crear un entorno colaborativo que cumpliera con las normas que impone ISO, y realmente así ha sido. Por lo tanto el resultado del proyecto ha sido beneficioso.

## 12.2 LÍNEAS FUTURAS

Como se viene comentando, la robótica colaborativa se puede decir que acaba de nacer y por lo tanto tiene mucho futuro por delante y puede evolucionar de una manera exponencial. El mundo de la industria está apostando muy fuerte por este concepto, y a corto plazo se notará un avance importante, sobre todo en la carga que puedan soportar estos robots.

El objetivo fundamental para el futuro del trabajo es poder crear a partir de un robot convencional, un robot colaborativo. Para ello se incorporarían los sensores en el cuerpo del robot para que pudieran tener conocimiento de todo el entorno de trabajo. A partir de ese momento, si hubiese riesgo de accidente los actuadores tendrían que intervenir a tiempo para evitar una colisión entre el robot y el trabajador.

Respecto a este proyecto también tiene capacidad de evolución. Si se quiere mejorar el tiempo de reacción de los sensores y los motores se podrían comprar unos de mejor calidad, sobre todo los sensores de contacto para que cuando notaran la colisión con el objeto la capacidad para reaccionar sea la mínima.

Otro método de mejorar las cintas transportadoras sería la de incorporar más sensores y de distintos tipos para que la seguridad sea mayor. Por ejemplo sensores de infrarrojos o sensores ópticos optimizarían el proyecto.



# ***BIBLIOGRAFÍA***



Universidad de Valladolid





## 13.1 REFERENCIAS RELACIONADAS CON LA ROBOTICA COLABORATIVA

- [1]. John R. Henry CPP, “Los cobots están dispuestos a trabajar contigo”.  
<http://www.plastico.com/temas/Los-cobots-estan-dispuestos-a-trabajar-contigo+108382>
- [2]. [http://www.elperiodicodearagon.com/noticias/robot-mata-trabajador-planta-volkswagen-alemania\\_1037459.html](http://www.elperiodicodearagon.com/noticias/robot-mata-trabajador-planta-volkswagen-alemania_1037459.html)
- [3]. <http://www.keyland.es/la-robotica-colaborativa/>
- [4]. Blog autómatas.  
<http://www.infoplcn.net/blogs-automatizacion/item/102143-robots-colaborativos>
- [5]. <http://www.infoplcn.net/actualidad-industrial/item/103207-iso-norma-ts15066-robots-colaborativos>
- [6]. José Luis Arribas Blanco, Auditor jefe de PRL, “Robots colaborativos y seguridad”  
<http://www.seguritecnia.es/seguridad-aplicada/industria/robots-colaborativos-y-seguridad>

## 13.2 REFERENCIAS RELACIONADAS CON LA SENSORIZACIÓN

- [7]. [http://www.ehu.eus/es/web/ingeniaritza-bilbao/content/-/asset\\_publisher/ZcG4/content/sensorizacion\\_avanzada\\_1415](http://www.ehu.eus/es/web/ingeniaritza-bilbao/content/-/asset_publisher/ZcG4/content/sensorizacion_avanzada_1415)
- [8]. <http://www.alcabot.com/alcabot/seminario2006/Trabajos/OlgaOlmedaOter.pdf>
- [9]. <http://www.monografias.com/trabajos31/transductores-sensores/transductores-sensores.shtml>
- [10]. Contaval, “¿Qué tipos de sensores fotoeléctricos existen?”.  
<http://www.contaval.es/que-tipos-de-sensores-fotoelectricos-existen/>
- [11]. Javier Rubio, “Sensores inductivos”.  
<https://www.seas.es/blog/automatizacion/sensores-inductivos/>
- [12]. <http://sensorstecnicas.net/es/productos/category/96/sensores-y-transmisores/sensores-capacitivos>



[13]. Diego Pérez de Diego, “Sensores de distancia por ultrasonidos”.  
[http://picmania.garcia-cuervo.net/recursos/redpictutorials/sensores/sensores de distancias con ultrasonidos.pdf](http://picmania.garcia-cuervo.net/recursos/redpictutorials/sensores/sensores_de_distancias_con_ultrasonidos.pdf)

[14]. <http://industria4.es/tag/smart-glasses/>

### 13.3 REFERENCIAS RELACIONADAS CON ARDUINO E IMPRESIÓN 3D

[15]. <http://arduino.cl/que-es-arduino/>

[16]. <http://industrino.es/historia-de-arduino/>

[17]. [http://intranet.bibliotecasgc.bage.es/intranet-tmpl/prog/local\\_repository/documents/17854.pdf](http://intranet.bibliotecasgc.bage.es/intranet-tmpl/prog/local_repository/documents/17854.pdf)

[18]. <https://www.prometec.net/funciones1/>



---

## CAPÍTULO 14

# ***ANEXOS***



Universidad de Valladolid



# I. PROGRAMACION SOFTWARE ARDUINO

## Control de velocidad y separación

```
const int EchoPin1 = 2;
const int TriggerPin1 = 3;
const int EchoPin2 = 4;
const int TriggerPin2 = 5;
const int pinENA = 6;
const int pinIN1 = 7;
const int pinIN2 = 8;
const int pinIN3 = 9;
const int pinIN4 = 10;
const int pinENB = 11;
const int ledRojo = 46;
const int ledAmarillo = 48;
const int ledVerde = 50;

int estado = 0;

const int waitTime = 2000; //espera entre fases
const int speed = 100; //velocidad de giro

const int pinMotorA[3] = { pinENA, pinIN1, pinIN2 };
const int pinMotorB[3] = { pinENB, pinIN3, pinIN4 };

void setup()
{
    Serial.begin(9600);

    pinMode(TriggerPin1, OUTPUT);
    pinMode(EchoPin1, INPUT);
    pinMode(TriggerPin2, OUTPUT);
    pinMode(EchoPin2, INPUT);
    pinMode(pinIN1, OUTPUT);
    pinMode(pinIN2, OUTPUT);
    pinMode(pinENA, OUTPUT);
    pinMode(pinIN3, OUTPUT);
    pinMode(pinIN4, OUTPUT);
    pinMode(pinENB, OUTPUT);
    pinMode(ledRojo, OUTPUT);
    pinMode(ledAmarillo, OUTPUT);
    pinMode(ledVerde, OUTPUT);
}

void loop()
{
    long tiempo1, distanciaCm1, tiempo2, distanciaCm2;

    digitalWrite(TriggerPin1, LOW); //para generar un pulso
limpio ponemos a LOW 4us
    delayMicroseconds(4);
    digitalWrite(TriggerPin1, HIGH); //generamos Trigger
(disparo) de 10us
    delayMicroseconds(10);
    digitalWrite(TriggerPin1, LOW);
```



```
    tiempo1 = pulseIn(EchoPin1, HIGH); //medimos el tiempo
entre pulsos, en microsegundos

    distanciaCm1 = tiempo1 * 10 / 292/ 2; //convertimos a
distancia, en cm
    digitalWrite(TriggerPin2, LOW); //para generar un pulso
limpio ponemos a LOW 4us
    delayMicroseconds(4);
    digitalWrite(TriggerPin2, HIGH); //generamos Trigger
(disparo) de 10us
    delayMicroseconds(10);
    digitalWrite(TriggerPin2, LOW);

    tiempo2 = pulseIn(EchoPin2, HIGH); //medimos el tiempo
entre pulsos, en microsegundos

    distanciaCm2 = tiempo2 * 10 / 292/ 2; //convertimos a
distancia, en cm

    if(estado == 0)
    {
        if((distanciaCm1 <= 75) && (distanciaCm1 > 30) ||
(distanciaCm2 <= 75) && (distanciaCm2 > 30))
        {
            digitalWrite(ledVerde,HIGH);
            digitalWrite(ledAmarillo,LOW);
            digitalWrite(ledRojo,LOW);

            Serial.print("Estoy entre 75 y 30 cm: ");
            Serial.println(distanciaCm2);

            avanzar(pinMotorA, 40);
            avanzar(pinMotorB, 250);
        }
        if((distanciaCm1 <= 30) && (distanciaCm1 > 12) ||
(distanciaCm2 <= 30) && (distanciaCm2 > 12))
        {
            digitalWrite(ledVerde,HIGH);
            digitalWrite(ledAmarillo,LOW);
            digitalWrite(ledRojo,LOW);

            Serial.print("Estoy entre 30 y 12 cm: ");
            Serial.println(distanciaCm2);

            avanzar(pinMotorA, 35);
            avanzar(pinMotorB, 180);
        }
        if((distanciaCm1 <= 12) && (distanciaCm1 > 5) ||
(distanciaCm2 <= 12) && (distanciaCm2 > 5))
        {
            digitalWrite(ledVerde,LOW);
            digitalWrite(ledAmarillo,HIGH);
            digitalWrite(ledRojo,LOW);

            Serial.print("Estoy entre 12 y 5 cm: ");
            Serial.println(distanciaCm2);

            avanzar(pinMotorA, 30);
```



```
    avanzar(pinMotorB, 150);
}
if((distanciaCm1 <= 5) || (distanciaCm2 <= 5))
{
    digitalWrite(ledVerde,LOW);
    digitalWrite(ledAmarillo,LOW);
    digitalWrite(ledRojo,HIGH);

    Serial.print("Estoy entre 5 y 0 cm: ");
    Serial.println(distanciaCm2);

    parada(pinMotorA);
    parada(pinMotorB);

    estado++;
    delay(1500);
}
}else
{
    if((distanciaCm1 >= 3) && (distanciaCm1 < 12) ||
(distanciaCm2 >= 3) && (distanciaCm2 < 12))
    {
        digitalWrite(ledVerde,LOW);
        digitalWrite(ledAmarillo,LOW);
        digitalWrite(ledRojo,HIGH);

        Serial.print("Estoy entre 3 y 12 cm: ");
        Serial.println(distanciaCm2);

        retroceder(pinMotorA, 30);
        retroceder(pinMotorB, 180);
    }
    if((distanciaCm1 >= 12) && (distanciaCm1 < 30) ||
(distanciaCm2 >= 12) && (distanciaCm2 < 30))
    {
        digitalWrite(ledVerde,LOW);
        digitalWrite(ledAmarillo,HIGH);
        digitalWrite(ledRojo,LOW);

        Serial.print("Estoy entre 12 y 30 cm: ");
        Serial.println(distanciaCm2);

        retroceder(pinMotorA, 35);
        retroceder(pinMotorB, 210);
    }
    if((distanciaCm1 >= 30) && (distanciaCm1 < 43) ||
(distanciaCm2 >= 30) && (distanciaCm2 < 43))
    {
        digitalWrite(ledVerde,HIGH);
        digitalWrite(ledAmarillo,LOW);
        digitalWrite(ledRojo,LOW);

        Serial.print("Estoy entre 30 y 47 cm: ");
        Serial.println(distanciaCm2);

        retroceder(pinMotorA, 40);
        retroceder(pinMotorB, 250);
    }
    if((distanciaCm1 >= 43) || (distanciaCm2 >= 43))
```



```
    {
        digitalWrite(ledVerde, HIGH);
        digitalWrite(ledAmarillo, LOW);
        digitalWrite(ledRojo, LOW);

        Serial.print("Estoy entre 47 y max: ");
        Serial.println(distanciaCm2);

        parada(pinMotorA);
        parada(pinMotorB);

        estado--;
        delay(3000);
    }
}

void avanzar(const int pinMotor[3], int speed)
{
    digitalWrite(pinMotor[1], HIGH);
    digitalWrite(pinMotor[2], LOW);

    analogWrite(pinMotor[0], speed);
}

void retroceder(const int pinMotor[3], int speed)
{
    digitalWrite(pinMotor[1], LOW);
    digitalWrite(pinMotor[2], HIGH);

    analogWrite(pinMotor[0], speed);
}

void parada(const int pinMotor[3])
{
    digitalWrite(pinMotor[1], LOW);
    digitalWrite(pinMotor[2], LOW);

    analogWrite(pinMotor[0], 0);
}
```



## Limitación de fuerza y energía

```
const int EchoPin = 2;
const int TriggerPin = 3;
const int EchoPin = 4;
const int TriggerPin = 5;
const int pinENA = 6;
const int pinIN1 = 7;
const int pinIN2 = 8;
const int pinIN3 = 9;
const int pinIN4 = 10;
const int pinENB = 11;

const int speed = 100;          //velocidad de giro

const int pinMotorA[3] = { pinENA, pinIN1, pinIN2 };
const int pinMotorB[3] = { pinENB, pinIN3, pinIN4 };

int LEDpin = 46;                // connect Red LED to pin 8 (PWM pin)
int fsrPinAnalog = A0;         // FSR está conectado al pin analigico0
int fsrReading;

int estado = 0;

void setup()
{
  Serial.begin(9600);

  pinMode(TriggerPin, OUTPUT);
  pinMode(EchoPin, INPUT);
  pinMode(pinIN1, OUTPUT);
  pinMode(pinIN2, OUTPUT);
  pinMode(pinENA, OUTPUT);
  pinMode(pinIN3, OUTPUT);
  pinMode(pinIN4, OUTPUT);
  pinMode(pinENB, OUTPUT);
  pinMode(LEDpin, OUTPUT);
}

void loop()
{
  long tiempo1, distanciaCm1;

  digitalWrite(TriggerPin1, LOW); //para generar un pulso limpio
  ponemos a LOW 4us
  delayMicroseconds(4);
  digitalWrite(TriggerPin1, HIGH); //generamos Trigger (disparo)
  de 10us
  delayMicroseconds(10);
  digitalWrite(TriggerPin1, LOW);

  tiempo1 = pulseIn(EchoPin1, HIGH); //medimos el tiempo entre
  pulsos, en microsegundos

  distanciaCm1 = tiempo1 * 10 / 292 / 2; //convertimos a
  distancia, en cm
```



```
    digitalWrite(TriggerPin2, LOW);           //para generar un pulso
limpio ponemos a LOW 4us
    delayMicroseconds(4);
    digitalWrite(TriggerPin2, HIGH);         //generamos Trigger
(disparo) de 10us
    delayMicroseconds(10);
    digitalWrite(TriggerPin2, LOW);

    tiempo2 = pulseIn(EchoPin2, HIGH);       //medimos el tiempo
entre pulsos, en microsegundos

    distanciaCm2 = tiempo2 * 10 / 292 / 2;    //convertimos a
distancia, en cm

    fsrReading = analogRead(fsrPinAnalog);
    int num = fsrReading/5;
    Serial.print(" Valor analógico = ");
    Serial.println(num);

    if(estado == 0)
    {
        if(num < 10)
        {
            digitalWrite(LEDpin, LOW);

            avanzar(pinMotorA, 40);
            avanzar(pinMotorB, 180);

        }else
        {
            digitalWrite(LEDpin, HIGH);

            parada(pinMotorA);
            parada(pinMotorB);

            estado++;
            delay(1500);
        }
    }else
    {
        if((distanciaCm1 > 0 && distanciaCm1 <= 43) || (distanciaCm2 >
0 && distanciaCm2 <= 43))
        {
            digitalWrite(LEDpin, LOW);

            retroceder(pinMotorA, 40);
            retroceder(pinMotorB, 180);

        }

        if((distanciaCm1 > 43) || (distanciaCm2 > 43))
        {

            parada(pinMotorA);
            parada(pinMotorB);

            estado--;
            delay(3000);
        }
    }
}
```



```
    }  
  }  
}  
  
void avanzar(const int pinMotor[3], int speed)  
{  
  digitalWrite(pinMotor[1], HIGH);  
  digitalWrite(pinMotor[2], LOW);  
  
  analogWrite(pinMotor[0], speed);  
}  
  
void retroceder(const int pinMotor[3], int speed)  
{  
  digitalWrite(pinMotor[1], LOW);  
  digitalWrite(pinMotor[2], HIGH);  
  
  analogWrite(pinMotor[0], speed);  
}  
  
void parada(const int pinMotor[3])  
{  
  digitalWrite(pinMotor[1], LOW);  
  digitalWrite(pinMotor[2], LOW);  
  
  analogWrite(pinMotor[0], 0);  
}
```



Universidad de Valladolid



## II. DATASHEET

### HC-SR04 Ultrasonic Range Finder

# Manual

#### Features

- Distance measurement range: 2cm • 400cm
- Accuracy: 0.3cm
- Detect angle: 15 degree
- Single +5V DC operation
- Current consumption: 15mA



Fig. 1

#### How It Works

HC-SR04 consists of ultrasonic transmitter, receiver, and control circuits. When triggered it sends out a series of 40KHz ultrasonic pulses and receives echo from an object. The distance between the unit and the object is calculated by measuring the traveling time of sound and output it as the width of a TTL pulse.

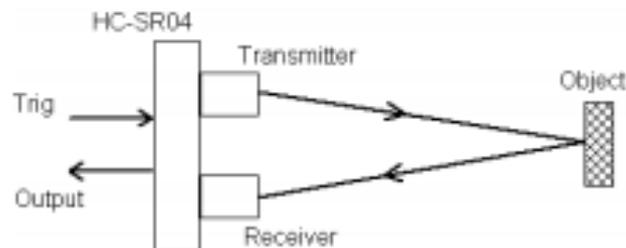


Fig. 2

#### How To Use It

To measure distance you need to generate a trig signal and drive it to the Trig Input pin. The trig signal level must meet TTL level requirements (i.e. High level > 2.4V, low level < 0.8V) and its width must be greater than 10us. At the same time you need to monitor the Output pin by measuring the pulse width of output signal. The detected distance can be calculated by the formula below.

$$\text{Distance} = \frac{\text{Pulse Width} * \text{Sound Speed}}{2}$$

where the pulse width is in unit of second and sound speed is in unit of meter/second. Normally sound speed is 340m/s under room temperature.

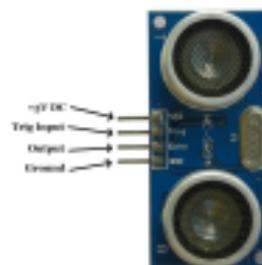


Fig.3

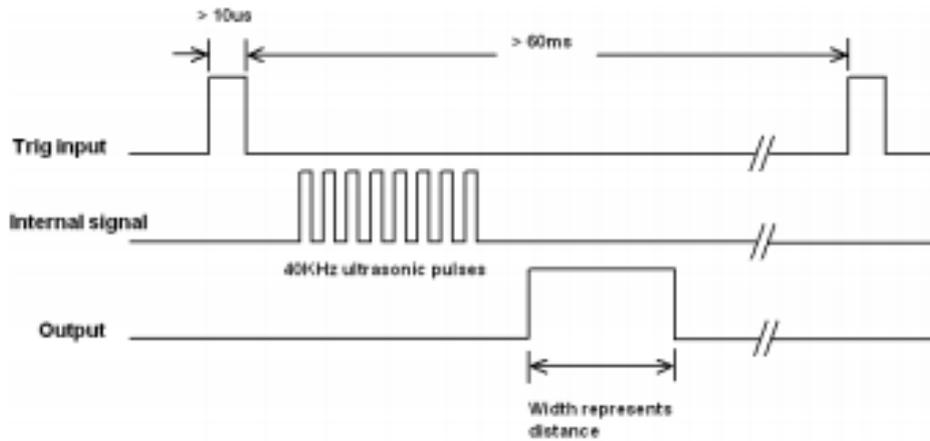


Fig. 4

- Notes:**
1. The width of trig signal must be greater than 10us
  2. The repeat interval of trig signal should be greater than 60ms to avoid interference between connective measurements.

### Specifications

Parameters	Specification
Operating Voltage	+5V DC
Operating Current	15mA
Operating Frequency	40KHz
Maximum Distance	400cm
Minimum Distance	2cm
Detect Angle	15 degree
Resolution	0.3cm
Input Trig Signal	>10us TTL pulse
Output Signal	TTL pulse with width representing distance
Weight	
Dimension	45 x 20 x 15 mm



## Sensor capacitivo touchless VMA 305

# velleman®

## VMA305

---

CAPACITIVE TOUCH SENSOR SWITCH



USER MANUAL



## USER MANUAL

### 1. Introduction

To all residents of the European Union

#### Important environmental information about this product



This symbol on the device or the package indicates that disposal of the device after its lifecycle could harm the environment. Do not dispose of the unit (or batteries) as unsorted municipal waste; it should be taken to a specialized company for recycling. This device should be returned to your distributor or to a local recycling service. Respect the local environmental rules.

**If in doubt, contact your local waste disposal authorities.**

Thank you for choosing Velleman®! Please read the manual thoroughly before bringing this device into service. If the device was damaged in transit, do not install or use it and contact your dealer.

### 2. Safety Instructions



- This device can be used by children aged from 8 years and above, and persons with reduced physical, sensory or mental capabilities or lack of experience and knowledge if they have been given supervision or instruction concerning the use of the device in a safe way and understand the hazards involved. Children shall not play with the device. Cleaning and user maintenance shall not be made by children without supervision.



- Indoor use only.  
Keep away from rain, moisture, splashing and dripping liquids.

### 3. General Guidelines



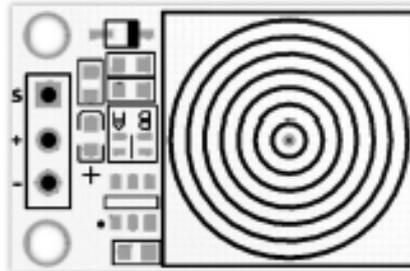
- Refer to the Velleman® Service and Quality Warranty on the last pages of this manual.
- Familiarise yourself with the functions of the device before actually using it.
- All modifications of the device are forbidden for safety reasons. Damage caused by user modifications to the device is not covered by the warranty.
- Only use the device for its intended purpose. Using the device in an unauthorised way will void the warranty.
- Damage caused by disregard of certain guidelines in this manual is not covered by the warranty and the dealer will not accept responsibility for any ensuing defects or problems.
- Nor Velleman nv nor its dealers can be held responsible for any damage (extraordinary, incidental or indirect) – **of any nature (financial, physical...)** arising from the possession, use or failure of this product.
- Due to constant product improvements, the actual product appearance might differ from the shown images.
- Product images are for illustrative purposes only.
- Do not switch the device on immediately after it has been exposed to changes in temperature. Protect the device against damage by leaving it switched off until it has reached room temperature.
- Keep this manual for future reference.

VMA305

4. Overview

A capacitive sensor is a touch sensor that requires little or no force to activate. Detects body capacitive touch, outputting analogue voltage.

Arduino®	VMA305
AO	S
+5 V	+
GND	-



connection .....	3 pins, GND - VCC - signal
voltage .....	3.3-5 VDC
dimensions .....	30 x 16 x 6 mm
weight .....	5 g

Use this device with original accessories only. Velleman nv cannot be held responsible in the event of damage or injury resulting from (incorrect) use of this device. For more info concerning this product and the latest version of this manual, please visit our website [www.velleman.eu](http://www.velleman.eu). The information in this manual is subject to change without prior notice.

© COPYRIGHT NOTICE

The copyright to this manual is owned by Velleman nv. All worldwide rights reserved. No part of this manual may be copied, reproduced, translated or reduced to any electronic medium or otherwise without the prior written consent of the copyright holder.



## Velleman® Service and Quality Warranty

Since its foundation in 1972, Velleman® acquired extensive experience in the electronics world and currently distributes its products in over 85 countries.

All our products fulfil strict quality requirements and legal stipulations in the EU. In order to ensure the quality, our products regularly go through an extra quality check, both by an internal quality department and by specialized external organisations. If, all precautionary measures notwithstanding, problems should occur, please make appeal to our warranty (see guarantee conditions).

### General Warranty Conditions Concerning Consumer Products (for EU):

• All consumer products are subject to a 24-month warranty on production flaws and defective material as from the original date of purchase.

• Velleman® can decide to replace an article with an equivalent article, or to refund the retail value totally or partially when the complaint is valid and a free repair or replacement of the article is impossible, or if the expenses are out of proportion.

You will be delivered a replacing article or a refund at the value of 100% of the purchase price in case of a flaw occurred in the first year after the date of purchase and delivery, or a replacing article at 50% of the purchase price or a refund at the value of 50% of the retail value in case of a flaw occurred in the second year after the date of purchase and delivery.

#### • Not covered by warranty:

- all direct or indirect damage caused after delivery to the article (e.g. by oxidation, shocks, falls, dust, dirt, humidity...), and by the article, as well as its contents (e.g. data loss), compensation for loss of profits;

- consumable goods, parts or accessories that are subject to an aging process during normal use, such as batteries (rechargeable, non-rechargeable, built-in or replaceable), lamps, rubber parts, drive belts... (unlimited list);

- flaws resulting from fire, water damage, lightning, accident, natural disaster, etc....;

- flaws caused deliberately, negligently or resulting from improper handling, negligent maintenance, abusive use or use contrary to the manufacturer's instructions;

- damage caused by a commercial, professional or collective use of the article (the warranty validity will be reduced to six (6) months when the article is used professionally);

- damage resulting from an inappropriate packing and shipping of the article;

- all damage caused by modification, repair or alteration performed by a third party without written permission by Velleman®.

• Articles to be repaired must be delivered to your Velleman® dealer, solidly packed (preferably in the original packaging), and be completed with the original receipt of purchase and a clear flaw description.

• Hint: In order to save on cost and time, please reread the manual and check if the flaw is caused by obvious causes prior to presenting the article for repair. Note that returning a non-defective article can also involve handling costs.

• Repairs occurring after warranty expiration are subject to shipping costs.

• The above conditions are without prejudice to all commercial warranties.

**The above enumeration is subject to modification according to the article (see article's manual).**

## Sensor de fuerza resistivo FSR 402



# FSR® 400 Series Data Sheet

Force Sensing Resistors®

### Features and Benefits

- Actuation force as low as 0.2N and sensitivity range to 20N
- Cost effective
- Ultra thin
- Robust; up to 10M actuations
- Simple and easy to integrate

### Description

Interlink Electronics FSR® 400 Series is part of the single zone Force Sensing Resistor® family. Force Sensing Resistors, or FSR's, are robust polymer thick film (PTF) devices that exhibit a decrease in resistance with increase in force applied to the surface of the sensor. This force sensitivity is optimized for use in human machine interface devices including automotive electronics, medical systems, industrial controls and robotics.

The FSR 400 Series sensors come in seven different models with four different connecting options. A battery operated demo is available. Call us for more information at +1 805-484-8855.



PN: PDS-10004-C

Human-Machine Interface Solutions for a Connected World



# FSR® 400 Series Data Sheet

Force Sensing Resistors\*

## Device Characteristics

Actuation Force*	~0.2N min
Force Sensitivity Range*	~0.2N – 20N
Force Resolution	Continuous (analog)
Force Repeatability Single Part	+/- 2%
Force Repeatability Part to Part	+/- 6% (Single Batch)
Non-Actuated Resistance	>10 Mohms
Hysteresis	+10% Average (RF+ - RF-)/RF+
Device Rise Time	< 3 Microseconds
Long Term Drift 1kg load, 35 days	< 5% log10(time)
Operating Temperature Performance Cold: -40°C after 1 hour Hot: +85°C after 1 hour Hot Humid: +85°C 95RH after 1 hour	-5% average resistance change -15% average resistance change +10% average resistance change
Storage Temperature Performance Cold: -25°C after 120 hours Hot: +85°C after 120 hours Hot Humid: +85°C 95RH after 240 hours	-10% average resistance change -5% average resistance change +30% average resistance change
Tap Durability Tested to 10 Million actuations, 1kg, 4Hz	-10% average resistance change
Standing Load Durability 2.5kg for 24 hours	-5% average resistance change
EMI	Generates no EMI
ESD	Not ESD Sensitive
UL	All materials UL grade 94 V-1 or better
RoHS	Compliant

\*Specifications are derived from measurements taken at 1000 grams, and are given as (one standard deviation/mean), unless otherwise noted. †Typical value. Force dependent on actuation interface, mechanics, and measurement electronics.

# FSR® 400 Series Data Sheet

Force Sensing Resistor®

## Connector Information

Bare Tail



Female Tin Contacts  
PN: TE 2-487406-4



Female Tin Contacts with 2 Pin Housing  
PN: TE 2-487406-4  
PH: TE 487378-1



Solder Tabs  
PN: TE 1-88997-2



Other Available Part Numbers:  
Hardware Development Kit, PN: 54-76247

## Application Information

For specific application needs please contact Interlink Electronics support team. An Integration Guide and Hardware Development Kit (HDK) are also available. FSR's are two-wire devices with a resistance that depends on applied force. Below is a force vs. resistance graph that illustrates a typical FSR® response characteristic. Please note that the graph values are reference only and actual values are dependent upon actuation system mechanics and sensor geometry.

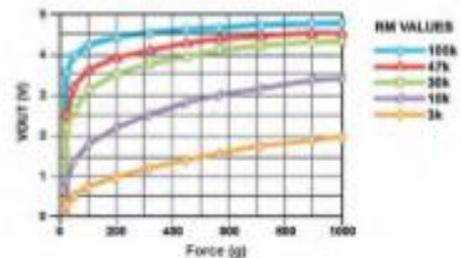
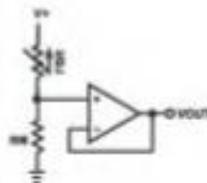
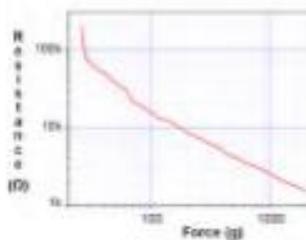
For simple force-to-voltage conversion, the FSR device is tied to a measuring resistor in a voltage divider (see figure below) and the output is described by the following equation.

$$V_{OUT} = \frac{R_M V_+}{(R_M + R_{FSR})}$$

In the configuration shown, the output voltage increases with increasing force. If  $R_{FSR}$  and  $R_M$  are swapped, the output swing will decrease with increasing force. The measuring resistor,  $R_M$ , is chosen to maximize the desired force sensitivity range and to limit current. Depending on the impedance requirements of the measuring circuit, the voltage divider could be followed by an op-amp.

A family of force vs.  $V_{OUT}$  curves is shown on the graph below for a standard FSR in a voltage divider configuration with various  $R_M$  resistors. A  $V_+$  of 5V was used for these examples. Please note that the graph values are for reference only and will vary between different sensors and applications.

Refer to the FSR Integration Guide for more integration methods and techniques.



# FSR® Model 402

Force Sensing Resistor\*

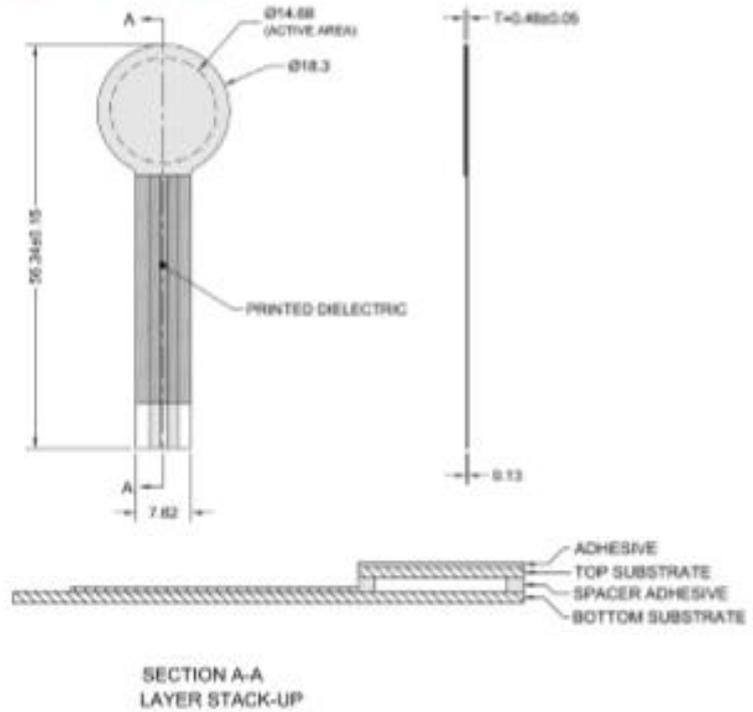
## Model 402:

Active Area:  $\varnothing 14.85\text{mm}$   
Nominal Thickness: 0.45mm  
Switch Travel: 0.15mm

## Available Part Numbers:

- PN: 44-29103 Model 402
  - No contacts or solder tabs
- PN: 24-00012 Model 402
  - with female contacts
- PN: 24-00001 Model 402
  - with female contacts and housing
- PN: 20-51794 Model 402
  - with solder tabs

## Sensor Mechanical Data



## Exploded View



# Motor Faulhaber

**FAULHABER**

## DC-Micromotors

Graphite Commutation

50 mNm

For combination with  
Gearheads:  
30T(S), 32T(S), 38T(S), 38T(S), 38A  
Encoders:  
HEDL 5540, HEDM 5500, HEDS 5500, HEDS 5540

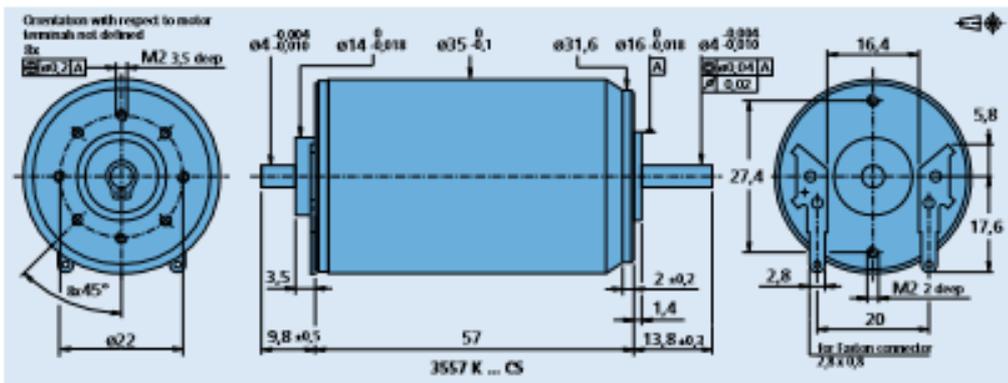
### Series 3557 ... CS

	3557 K	001 G5	012 G5	020 G5	022 G5	032 G5	
1 Nominal voltage	U <sub>N</sub>	9	12	20	24	48	V
2 Terminal resistance	R	0,7	1,34	4	5,5	23	Ω
3 Output power	P <sub>max</sub>	28,1	26,1	24,3	25,4	24,1	W
4 Efficiency, max.	η <sub>max</sub>	78	79	79	78	76	%
5 No-load speed	n <sub>0</sub>	5 700	5 400	5 500	5 500	5 200	rpm
6 No-load current (with shaft ø 4 mm)	I <sub>0</sub>	0,19	0,125	0,07	0,065	0,04	A
7 Stall torque	M <sub>st</sub>	188	185	169	176	177	mNm
8 Friction torque	M <sub>f</sub>	2,8	2,6	2,4	2,7	3,5	mNm
9 Speed constant	k <sub>v</sub>	643	456	279	233	110	rpm/V
10 Back-EMF constant	k <sub>e</sub>	1,56	2,79	3,59	4,3	9,05	mV/spm
11 Torque constant	k <sub>t</sub>	14,9	20,9	34,2	41	86,5	mNm/A
12 Current constant	k <sub>i</sub>	0,067	0,048	0,029	0,024	0,012	A/mNm
13 Slope of n-M curve	Δn/ΔM	30,3	29,2	32,5	31,3	29,4	rpm/mNm
14 Rotor inductance	L	100	220	630	850	3 400	μH
15 Mechanical time constant	T <sub>m</sub>	16	16	16	16	16	ms
16 Rotor inertia	J	50	52	47	49	52	gcm <sup>2</sup>
17 Angular acceleration	α <sub>max</sub>	37	35	36	36	34	10 <sup>3</sup> rad/s <sup>2</sup>
18 Thermal resistance	R <sub>th</sub> / R <sub>th</sub> 1	1,5 / 9					K/W
19 Thermal time constant	τ <sub>th</sub> / τ <sub>th</sub> 1	15 / 900					s
20 Operating temperature range:							
- motor		-30 ... +125					°C
- rotor, max. permissible		+155					°C
21 Shaft bearings		ball bearings, preloaded					
22 Shaft load max.:							
- with shaft diameter		4					mm
- radial at 3 000 rpm (3 mm from bearing)		30					N
- axial at 3 000 rpm		5					N
- axial at standstill		50					N
23 Shaft play							
- radial	e	0,015					mm
- axial	f	0					mm
24 Housing material		steel, zinc galvanized and passivated					
25 Weight		275					g
26 Direction of rotation		clockwise, viewed from the front face					

Recommended values - mathematically independent of each other

27 Speed up to	n <sub>max</sub>	5 000	5 000	5 000	5 000	5 000	rpm
28 Torque up to <sup>1)</sup>	M <sub>max</sub>	50	50	50	50	50	mNm

<sup>1)</sup> thermal resistance R<sub>th</sub> by 40% reduced



For notes on technical data and lifetime performance refer to "Technical Information".  
Edition 2012 - 2013

© DR. HEIZ FAULHABER GMBH & CO. KG  
Specifications subject to change without notice.  
www.faulhaber.com

## Driver L293B



**L293B**  
**L293E**

### PUSH-PULL FOUR CHANNEL DRIVERS

- OUTPUT CURRENT 1A PER CHANNEL
- PEAK OUTPUT CURRENT 2A PER CHANNEL (non repetitive)
- INHIBIT FACILITY
- HIGH NOISE IMMUNITY
- SEPARATE LOGIC SUPPLY
- OVERTEMPERATURE PROTECTION

#### DESCRIPTION

The L293B and L293E are quad push-pull drivers capable of delivering output currents to 1A per channel. Each channel is controlled by a TTL-compatible logic input and each pair of drivers (a full bridge) is equipped with an inhibit input which turns off all four transistors. A separate supply input is provided for the logic so that it may be run off a lower voltage to reduce dissipation.

Additionally, the L293E has external connection of sensing resistors, for switchmode control.

The L293B and L293E are package in 16 and 20-pin plastic DIPs respectively ; both use the four center pins to conduct heat to the printed circuit board.



DIP16

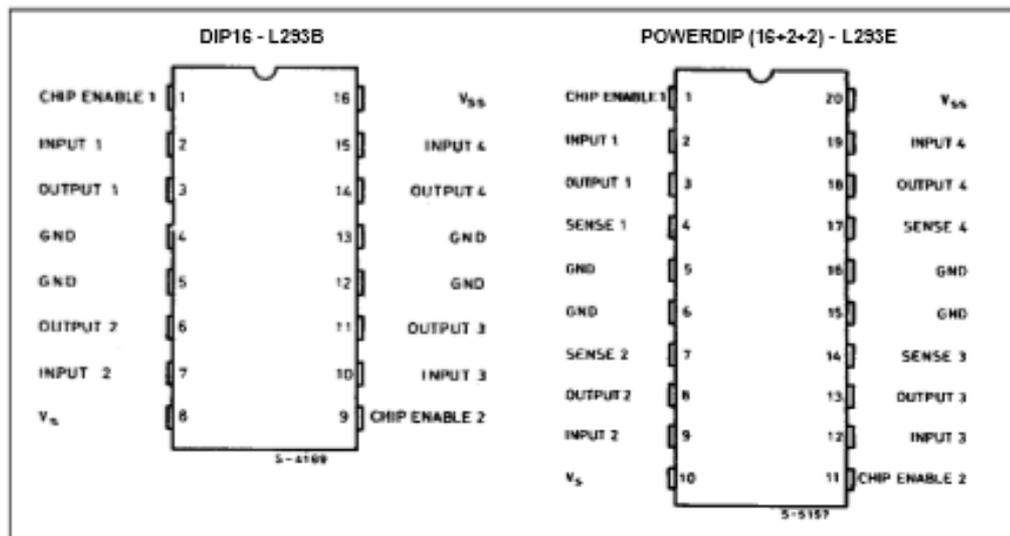
ORDERING NUMBER : L293B



POWERDIP (16 + 2 + 2)

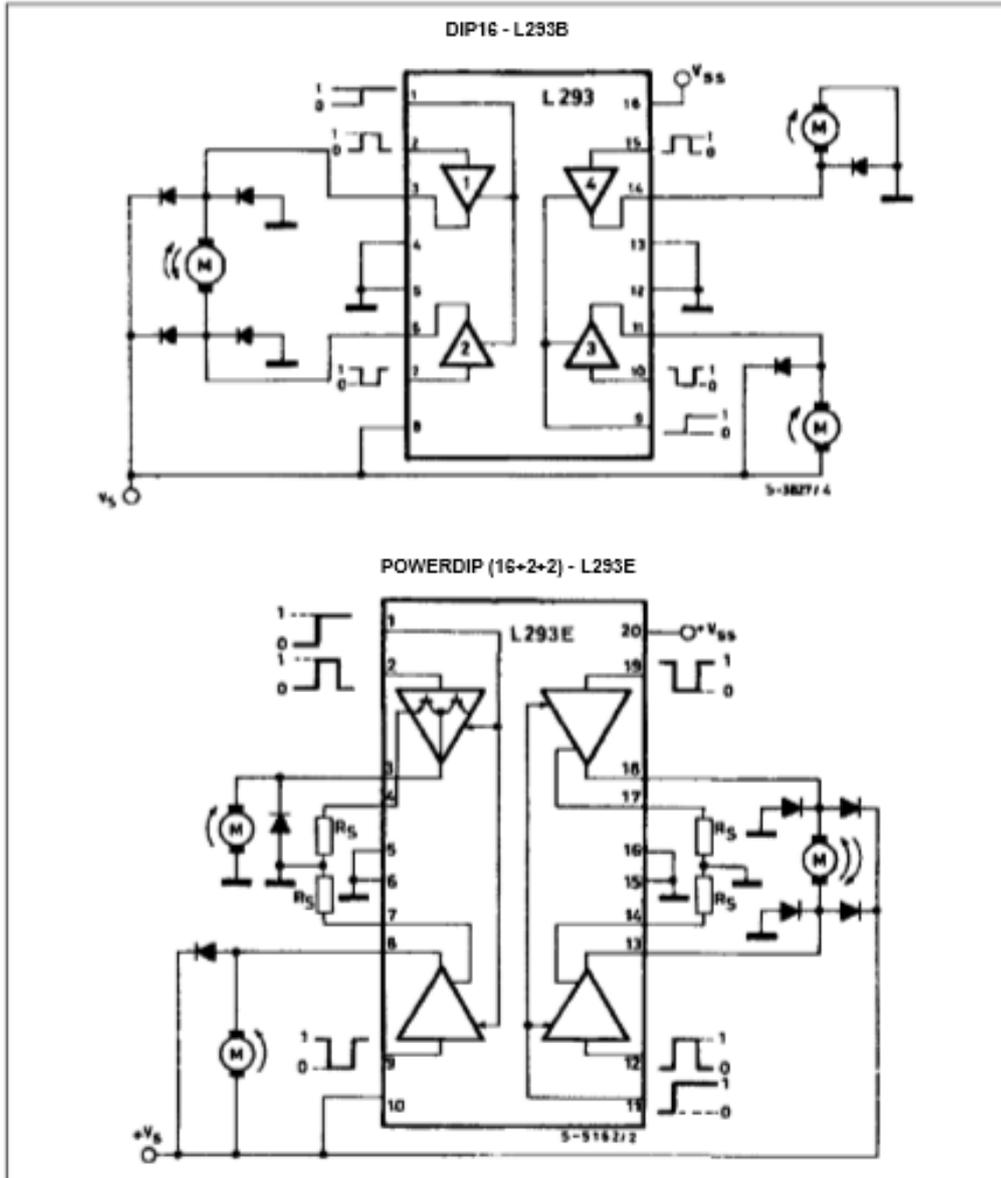
ORDERING NUMBER : L293E

#### PIN CONNECTIONS



### L293B - L293E

#### BLOCK DIAGRAMS





Universidad de Valladolid

