



UNIVERSIDAD TECNOLÓGICA ISRAEL

ESCUELA DE POSGRADOS "ESPOG"

MAESTRÍA EN

MENCIÓN: ELECTRÓNICA Y AUTOMATIZACIÓN

Resolución: RPC-SO-09-No.265-2021

PROYECTO DE TITULACIÓN EN OPCIÓN AL GRADO DE MAGISTER

Título del proyecto:
Sistema de Control y Monitoreo de Brazo Robótico y Banda Transportadora para el Laboratorio de Electrónica de la Universidad Tecnológica Israel mediante IoT basado en la plataforma Blynk
Línea de Investigación:
Ciencias de la ingeniería aplicadas a la producción, sociedad y desarrollo sustentable.
Campo amplio de conocimiento:
Ingeniería, industria y comunicación
Autor:
Boris Santiago Chilán Saltos
Tutores:
PHD. Maryory Urdaneta Mg. Wilmer Albarracín

Quito – Ecuador

2024

APROBACIÓN DEL TUTOR



Yo, PhD Maryory Urdaneta con C.I.:1759316126 en mi calidad de Tutor del proyecto de investigación titulado: **Sistema de Control y Monitoreo de Brazo Robótico y Banda Transportadora para el Laboratorio de Electrónica de la Universidad Tecnológica Israel mediante IoT basado en la plataforma Blynk.**

Elaborado por: **Boris Santiago Chilán Saltos**, de C.I.: **0908383417**, estudiante de la Maestría: **ELECTRÓNICA Y AUTOMATIZACIÓN de la UNIVERSIDAD DE ISRAEL (UISRAEL)**, como parte de los requisitos sustanciales con fines de obtener el Título de Magister, me permito declarar que luego de haber orientado, analizado y revisado el trabajo de titulación, lo apruebo en todas sus partes.

Quito D.M., 22 de Marzo de 2024

Firma

DECLARACIÓN DE AUTORIZACIÓN POR PARTE DEL ESTUDIANTE



Yo, **Boris Santiago Chilán Saltos** con C.I: **0908383417**, autor/a del proyecto de titulación denominado: **Sistema de Control y Monitoreo integrado de Brazo Robótico y Banda Transportadora para uso del Laboratorio de Electrónica de la Universidad Tecnológica Israel mediante IoT basado en la plataforma Blynk**. Previo a la obtención del título de **Magister en Electrónica y Automatización**.

1. Declaro tener pleno conocimiento de la obligación que tienen las instituciones de educación superior, de conformidad con el Artículo 144 de la Ley Orgánica de Educación Superior, de entregar el respectivo trabajo de titulación para que sea integrado al Sistema Nacional de Información de la Educación Superior del Ecuador para su difusión pública respetando los derechos de autor.
2. Manifiesto mi voluntad de ceder a la Universidad Tecnológica Israel los derechos patrimoniales consagrados en la Ley de Propiedad Intelectual del Ecuador, artículos 4, 5 y 6, en calidad de autor@ del trabajo de titulación, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente. En concordancia suscribo este documento en el momento que hago entrega del trabajo final en formato impreso y digital como parte del acervo bibliográfico de la Universidad Tecnológica Israel.
3. Autorizo a la SENESCYT a tener una copia del referido trabajo de titulación, con el propósito de generar un repositorio que democratice la información, respetando las políticas de prosperidad intelectual vigentes.

Quito D.M., 22 de Marzo de 2024

Firma

Tabla de contenidos

APROBACIÓN DEL TUTOR	1
DECLARACION DE AUTORIZACIÓN POR PARTE DEL ESTUDIANTE	2
Índice de tablas	5
INFORMACIÓN GENERAL	7
Contextualización del tema	7
Problema de investigación	8
Objetivo general	9
Objetivos específicos	10
Vinculación con la sociedad	10
CAPÍTULO I: DESCRIPCIÓN DEL PROYECTO	
1.1. Contextualización de fundamentos teóricos	11
1.2. Contextualización general del estado del arte	12
1.3 Proceso investigativo metodológico	13
CAPÍTULO II: PROPUESTA	14
2.1 Marco Teórico	14
2.2 Descripción del proyecto	14
2.2.1 Brazo robótico con seis grados de libertad arduino stm32 robot	15
2.2.2 Servomotor ks3620 digital servo	17
2.2.3 Banda transportadora de control de motor de paso: kit educational diy	18
2.2.4 Sensor de distancia: ir sensor switch e18-d80nk-n.	18
2.2.5 Motor de paso integrado: kh42 series hybrid type, 2 phase stepping motor (1.8 degree/step) - nema 17.	20
2.2.6 Comunicación del brazo robotico y banda transportadora	21

2.2.7	Freenove ultimate starter kit for esp32-wrover	21
2.2.8	A4988 stepper motor driver board	23
2.2.9	Aplicaciones del software blynk	24
2.2.10	Plataforma blynk console.	25
2.2.11	Software de programación arduino ide	26
2.3	Metodología de trabajo.	28
2.3.1	Estructura general	30
2.3.2	Configuración y programación pwm del controlador con el dispositivo Pca9685 para control del brazo robótico.	29
2.3.2.1	Conexión hardware	29
2.3.2.2	Teoría de programación pwm	29
2.3.3	Configuración y programación del dispositivo a4988 Para control de banda transportadora.	33
2.3.3.1	Conexión hardware	33
2.3.4	Programación blynk	35
2.4	Validación de la propuesta.	39
2.5	Matriz de articulación de la propuesta	48
2.6	Evidencia de Instalación de equipos	50
	CONCLUSIONES	50
	RECOMENDACIONES	51
	BILBIOGRAFIA	52
	ANEXOS	54

Índice de tablas

Tabla 1 <i>Tabla de dimensiones físicas del brazo robótico</i>	16
Tabla 2 <i>Datos técnicos importantes de los servos motores PWM</i>	17
Tabla 3 Especificaciones técnicas del motor de paso.....	20
Tabla 4 Datos técnicos del dispositivo integrado de potencia.....	23
Tabla 5 Plataforma de variables usadas en el proyecto en la plataforma Blynk.....	26
Tabla 6 Tabla de movimientos físicos de los servo motores.....	32
Tabla 7 Tabla de configuración de pines para movimiento del motor de paso.....	34
Tabla 8 Tabla de variables Blynk.....	38
Tabla 9 Perfil de validadores.....	40
Tabla 10 Criterios de evaluación	41
Tabla 11 Validador 1.....	42
Tabla 12 Escala de evaluación por Mgtr. Miguel Cedeño.....	42
Tabla 13 Validador 2.....	43
Tabla 14 Escala de evaluación por Mgtr. Roberto Pérez Suarez.....	43
Tabla 15 Validador 3.....	44
Tabla 16 Escala de evaluación por PHD. Maryory Urdaneta Herrera	44
Tabla 17 Matriz de articulación	45

Índice de figuras

Figura 1 Modelo de control de brazo y banda transportadora.....	14
Figura 2 Modelo general del proyecto para comunicaciones blynk y equipos.....	15
Figura 3 Brazo robótico con 6 grados de libertad y definición de posiciones servos.....	16
Figura 4 Servomotor dc de gran capacidad de torque.....	17
Figura 5 Banda transportadora con sus sensores de proximidad y motor de paso.....	18
Figura 6 Sensor infrarrojo y sus características de seteos.....	19
Figura 7 Sensor infrarrojo y sus características de seteos de aproximación.....	19
Figura 8 Diagrama eléctrico de conversión de señal de Vdc	20
Figura 9 Dimensiones físicas del motor de paso de la banda transportadora.....	20
Figura 10 Diagrama de enlace blynk con los dispositivos.....	21
Figura 11 Esquema diseño ESP32.....	22
Figura 12 Diseño del driver de potencia A4988.....	25
Figura 13 Dispositivo celular usando plataforma Blynk.....	24
Figura 14 Creación de control con dispositivo celular para automatización IoT.....	25
Figura 15 Plataforma de programación Arduino.....	27
Figura 16 Diagrama de flujo de tarea del brazo robótico y banda transportadora.....	28
Figura 17 Conexiones de los dispositivos con el driver de potencia y servos.....	29
Figura 18 . Lógica de programación PWM.....	30
Figura 19 Lógica de señales PWM.....	30
Figura 20 Montaje físico y conexiones del brazo robótico.....	33
Figura 21 Esquema de conexiones con la banda transportadora.....	33
Figura 22 Driver de potencia A4988 para el control de motor de paso.....	35
Figura 23 APP Blynk para dispositivos androide.....	35
Figura 24 Creación de usuario en la página de Blynk.....	36
Figura 25 Plataforma de imagen en el teléfono celular.....	39

INFORMACIÓN GENERAL

CONTEXTUALIZACIÓN DEL TEMA

La enseñanza para los estudiantes sobre la automatización y el control remoto de dispositivos físicos utilizando tecnología IoT es uno de los retos universitarios que presentan actualmente las instituciones educativas en todo el mundo, así como con el desarrollo de la industria 4.0.

El brazo robótico se diseñó para realizar tareas simples, como recoger y colocar objetos, por medio de un control realizado en forma virtual en el celular, la cual podrá modificar sus trayectorias de desplazamientos y velocidad cuando lo requiera el usuario, mientras que la banda transportadora se lo utilizará para el movimiento de los objetos de un lugar a otro, que a su vez tiene instalado un sensor infra rojo que indicará que hay un objeto a la vista y el brazo comience su trayectoria de embalaje de caja, contando con un contador para determinar el proceso de recolección de cajas.

La plataforma Blynk es una herramienta online que permite crear interfaces de usuario personalizadas para dispositivos conectados. Esto permitirá a los estudiantes controlar el brazo robótico y la banda transportadora a través de una aplicación móvil, lo que facilita la interacción y el aprendizaje práctico.

La cantidad de personas que pueden controlar el brazo robótico y la banda transportadora en un entorno educativo utilizando Blynk depende de varios factores, como la configuración del sistema, la capacidad de la red y la disponibilidad de dispositivos móviles.

En general, Blynk permite que múltiples usuarios accedan y controlen dispositivos conectados simultáneamente a través de la aplicación móvil. Sin embargo, la cantidad exacta de usuarios que pueden controlar el sistema al mismo tiempo puede estar limitada por factores como el ancho de banda de la red y la capacidad de procesamiento del servidor Blynk.

BENEFICIARIOS DIRECTOS

Los estudiantes que participan en el proyecto son los principales beneficiarios directos. Tendrán la oportunidad de aprender sobre programación, electrónica, mecánica y control de procesos a través de la interacción práctica con el brazo robótico y la banda transportadora.

Además, desarrollarán habilidades de resolución de problemas, pensamiento crítico y trabajo en equipo.

Los profesores que facilitan el proyecto también se benefician directamente al tener la oportunidad de enseñar, esto les permite involucrar a los estudiantes en un aprendizaje activo y fomentar su interés en áreas relacionadas con la tecnología y la ingeniería.

Los investigadores y desarrolladores que están involucrados en la creación y mejora de la plataforma Blynk y los componentes del proyecto, como el brazo robótico y la banda transportadora, también son beneficiarios directos. Su trabajo contribuye al avance de la tecnología educativa y proporciona herramientas más efectivas para la enseñanza y el aprendizaje.

BENEFICIARIOS INDIRECTOS

Las instituciones educativas, como escuelas, colegios y universidades, son beneficiarias indirectas del proyecto, ya que pueden mejorar su reputación al ofrecer programas educativos innovadores y de vanguardia.

La industria también puede beneficiarse indirectamente al contar con una fuerza laboral más capacitada en áreas relacionadas con la tecnología y la ingeniería. Los estudiantes que participan en proyectos como este pueden adquirir habilidades y conocimientos que son directamente aplicables en entornos industriales, lo que puede contribuir al crecimiento y la innovación en diferentes sectores.

La comunidad en general puede beneficiarse de manera indirecta al tener acceso a individuos mejor educados y capacitados en áreas de tecnología y ciencia. Esto puede llevar a un mayor desarrollo económico y social, así como a la creación de nuevas oportunidades de empleo y emprendimiento en la región.

PROBLEMA DE INVESTIGACIÓN

Uno de los problemas que enfrentan la mayoría de las universidades en Ecuador es la falta de recursos financieros y tecnológicos adecuados para mantenerse actualizado con las últimas tendencias y avances en el campo de la robótica.

Los laboratorios de robótica requieren equipos especializados, materiales y software para llevar a cabo investigaciones y proyectos. La falta de fondos suficientes puede dificultar la

adquisición de estos recursos, lo que limita las capacidades del laboratorio y restringe las oportunidades de investigación y desarrollo.

En el campo de la tecnología con respecto a la robótica avanza rápidamente, además los equipos y herramientas obsoletos pueden limitar la capacidad del laboratorio para realizar investigaciones de vanguardia y estar actualizado con las tecnologías emergentes.

Al mantener el estándar de enseñanza limitada con respecto a los componentes en el área de la robótica y automatización, se tendrá para el futuro un descenso en el interés de que los estudiantes puedan diseñar, programar e implementar nuevos diseños e ideas con respecto a la industria 4.0 usando en general las herramientas globales IoT.

La adquisición de equipos como brazos robóticos y bandas transportadoras puede ser costosa, lo que limita el acceso de algunas instituciones educativas a estas tecnologías. Esto puede dificultar la implementación de prácticas educativas efectivas en este campo.

Dado que la tecnología en el área de la robótica y la automatización está en constante evolución, puede ser difícil para las universidades mantener actualizados sus planes de estudio y currículos educativos para reflejar los últimos avances el área.

La robótica y la automatización son campos interdisciplinarios que requieren conocimientos en áreas como la ingeniería eléctrica, la ingeniería mecánica, la informática y la matemática. La falta de integración interdisciplinaria en la enseñanza de estos temas puede limitar la comprensión integral de los estudiantes.

Para abordar estos problemas, en la universidad de Israel podrían considerar implementar estrategias como la búsqueda de financiamiento para adquirir equipos, el desarrollo profesional para docentes, la revisión regular de los planes de estudio para mantenerlos actualizados, la promoción de la colaboración interdisciplinaria y la creación de un entorno inclusivo que fomente la interacción de todos los estudiantes.

OBJETIVO GENERAL

Desarrollar un sistema de control y monitoreo integrado de brazo robótico y banda transportadora para uso del Laboratorio de Electrónica de la Universidad Tecnológica Israel mediante IoT basado en la plataforma Blynk

OBJETIVOS ESPECÍFICOS

- Identificar los componentes necesarios para la implementación.
- Desarrollar el brazo robótico y la banda transportadora.
- Integrar la plataforma IoT Blynk para el monitoreo y control.
- Realizar pruebas de funcionamiento.

VINCULACIÓN CON LA SOCIEDAD Y BENEFICIARIOS DIRECTOS:

Las empresas que desarrollan tecnologías relacionadas con la automatización y la robótica pueden beneficiarse de la colaboración en proyectos educativos. Estas empresas pueden proporcionar apoyo técnico, acceso a equipos y recursos, e incluso oportunidades de pasantías o empleo para estudiantes que demuestren habilidades y conocimientos relevantes.

Los proyectos educativos que involucran brazos robóticos y sistemas de transporte automatizados preparan a los estudiantes para roles futuros en la industria. Esto puede ser especialmente beneficioso para las empresas que buscan contratar talento en áreas como la ingeniería robótica, la automatización industrial y el desarrollo de sistemas de control.

Pueden servir como vehículos para la transferencia de conocimiento a la comunidad. Esto puede incluir la organización de talleres, seminarios o eventos abiertos al público donde se comparta información sobre la tecnología y su aplicación en diversos contextos.

Puede ser aplicada en proyectos comunitarios. Por ejemplo, estos sistemas podrían utilizarse en la automatización de procesos en entornos agrícolas, la gestión de inventario en organizaciones sin fines de lucro o la optimización de la logística en empresas locales.

La colaboración en proyectos educativos de este tipo puede fomentar la innovación y mejorar la competitividad tanto para las empresas como para las comunidades locales. Al trabajar juntos en la aplicación práctica de tecnologías avanzadas, las empresas pueden encontrar nuevas oportunidades de negocio y las comunidades pueden beneficiarse del desarrollo económico y social que resulta de estas colaboraciones.

CAPÍTULO I: DESCRIPCIÓN DEL PROYECTO

1.1 Contextualización de fundamentos teóricos.

Para revisar algunas teorías usadas en el presente trabajo, se buscó tesis o investigación relacionada con modelos de robótica y automatización IoT en varias literaturas en repositorios de la universidad de Israel y universidades del país o externa.

Diseños y programación de plataforma de automatización IoT relacionado con la robótica y programación de software:

“El brazo robótico implementado permitirá que sus movimientos realizados sean en forma inalámbrica, dando como resultado una mayor flexibilidad en momento de su manipulación por la ausencia de cables logrando que el operario maniobre sus movimientos sin necesidad de una completa cercanía al robot”, (Verónica Alexandra Guilcaso Molina en enero de 2011). El aporte del proyecto principal es el uso de programación estándar y sus características de lógica de control con respecto un modelo industrial.

“Se basa en un diseño y desarrollo para un brazo robótico que utiliza servomotores Dynamixel; El software empleado para dicho sistema es Paython que es un lenguaje de programación libre, brinda al proyecto de fiabilidad y factibilidad, resolviendo el problema de financiamiento que presentan varios sistemas robóticos ya que los softwares que vienen con ellos presentan grandes inversiones. “(Sr.Javier Mamarandi, Latacunga 2014). El proyecto que se realiza en la plataforma Blynk, son las mismas características de un brazo robótico de 6 grados de libertad, se logró implementar una lógica de funcionamiento industrial con el empaquetamiento de cajas contadoras y sincronización de banda transportadora.

“La arquitectura de Cloud Computing aportó una reducción de recursos y costos ya que toda la infraestructura de la teleoperación se la implementó usando los APIs y Services de Google Cloud Platform y solo se requiere tener una cuenta de Gmail, dicha plataforma actúa como broker lo que permitió una interacción directa entre las herramientas de Firebase para el control de los mandos teleoperados por medio de dispositivos móviles y permite conectarse de forma remota a tarjetas de control que se conectan a la red de Internet en este caso se usó una ESP8266 y permitió controlar las acciones de teleoperación del brazo robótico a través del protocolos MQTT que es muy usado en aplicaciones IoT y para este sistema integrado en tiempo

real y se alcanzó obtener un menor uso de ancho de banda, baja latencia y mayor rendimiento. "(Juan David Chimarro Amaguaña, Quito 2017). Se uso la lógica y características de conexiones para aplicaciones IoT (Internet of Things), el cual se puede controlar y programar en tiempo real con aplicaciones de software libre.

"La conexión a Internet con una IP pública fija es un factor indispensable para la comunicación y verificación de los registros de la información almacenada y llevar estadísticas de las cantidades procesadas. " (Juan David Chimarro Amaguaña, Quito 2017). El uso de las aplicaciones y conexiones en tiempo real, se aplicó en el modelo de cajas empaquetadoras, el cual el usuario tiene la facilidad de cambiar parámetros de la cantidad de cajas que se pueda empaacar con un simple monitoreo contador online basado en la plataforma BLYNK.

1.2. Contextualización general del estado del arte

En este capítulo, se detallará los conceptos enlazados con los prototipos de un brazo robótico y manejo de banda transportadora para demostrar el uso de una implementación de una industria 4.0 basado en IoT con la plataforma BLYNK para el uso de los estudiantes en el laboratorio de electrónica de la Universidad Israel.

Con el pasar de los años, el concepto de usar plataformas IoT se hacen más comunes y eso permite que los profesores deban impartir clases más modernas y didácticas con el uso de implementación de equipos adaptables para el campo educativo.

Para mejorar el entendimiento una industria 4.0 impartido a los estudiantes, es necesario desarrollar una planta automatizada el cual se pueda usar plataformas de programación en la nube (cloud), donde pueda el usuario tener el control flexible del de los equipos y a la vez monitorear los estados de la maqueta de la planta industrial.

El laboratorio de electrónica de la universidad de Israel cuenta con instalaciones adecuadas para impartir clase destinado a la automatización, para lo cual se desea que se siga desarrollando en el área de la robótica y sus beneficios para impartir enseñanza en la industria 4.0.

Con base al desarrollo de la institución y proponer mejoras que puedan incrementar la capacidad de desarrollo en la automatización. Adicionando algunas modificaciones disponibles en la maqueta de la planta industrial, se podrá mejorar el entendimiento de la instrumentación y automatización industrial, el cual es el objetivo fundamental para que los estudiantes lo adquieran en la carrera universitaria y lo apliquen a la vida profesional.

1.3. Proceso investigativo metodológico

Para el proyecto de desarrollo para el área de la educación se desarrollará un método deductivo que se basa en la lógica formal y la inferencia para llegar a conclusiones específicas a partir de premisas generales o universales. Es un proceso que se utiliza ampliamente en la ciencia, las matemáticas, la filosofía y otros campos para validar teorías y hacer predicciones sobre el mundo real.

Como premisa mayor será establecer principios generales o leyes universales relacionadas con la robótica, la automatización y la programación de dispositivos IoT.

Como premisa menor es el de identificar características específicas del proyecto, como los componentes mecánicos y electrónicos disponibles, así como los objetivos educativos del proyecto. En este caso sería el brazo robótico de 6 grados de libertad y una banda transportadora de maqueta educativa.

Con todos estos elementos se ha recopilado datos, información y videos para que se pueda llevar un estándar de equipos factibles en la automatización para el campo educativo, además de poder entender los avances tecnológico que avancen cada vez más rápido en el campo de la automatización, especialmente en el área de las plataformas IoT, para el caso especial es el desarrollo en la plataforma BLYNK.

CAPITULO II: PROPUESTA

2.1 Marco teórico

A continuación, se mostrará la maqueta industrial de recolección de cajas y contador, plataformas de programación y lenguajes que se requieren específicamente en el desarrollo del proyecto.

2.2. Descripción del proyecto

La maqueta industrial 4.0 cuenta con un brazo mecánico de 6 grados de libertad y una banda transportadora, para lo cual, el control se basará bajo la plataforma BLYNK IoT.

Al tener 2 tipos de modos de control, el brazo robótico y la banda transportadora se lo realizará por comunicación wifi en la red de la nube (cloud), para lo cual se comunicará con los microcontroladores ESP32 Wrover y ESP32 S3 Wroom, lo cual permitirá el control de los accionamientos de brazos y banda, además de tener un sensor óptico de distancia que detectará la presencia de un objeto.

El modo de control manual permitirá al usuario tener el control total del brazo robótico y grabar las trayectorias específicas para que pueda desarrollar las tareas de repetición continua.

El modo de control automático solo se podrá acceder cuando se haya realizado la trayectoria del brazo robótico, una vez realizado la primera condición, la maqueta industrial procederá a funcionar automáticamente manipulando y trasladando las cajas contadoras, al mismo tiempo se podrá observar el funcionamiento de la banda transportadora.

El lenguaje base de programación es C el cual se lo desarrolla en el ambiente Arduino IDE 2.3.2, el cual se desarrolló principalmente una estructura de programación por estado FreeRTOS por núcleo (CORE0 y CORE1) para que realicen tareas independientes y no colapse el microcontrolador con el uso de tareas de WIFI y enlaces con la plataforma BLYNK

Figura 1

Modelo de control de brazo y banda transportadora

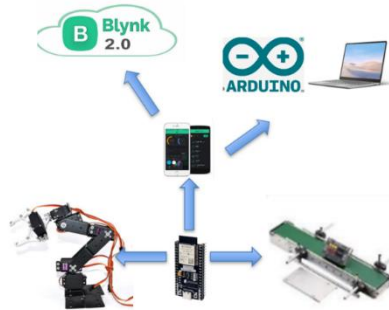


Nota. Esquemático de un brazo robótico y banda transportadora versión educativa, Amazon.com.mx,2024 (<https://www.amazon.com.mx/Dobot-Educativo-educativa-transportadora-Edu-Magician-Belt/dp/B07PPDFST>)

Se muestra la maqueta del brazo robótico y la banda transportadora automatizada, en cual se observa los componentes del sistema aplicado al proyecto.

Figura 2

Modelo general del proyecto para comunicaciones blynk y equipos



Nota. *Diseño del proyecto para enlace de conexión cloud BLYNK, Arduino, brazo robótico y banda transportadora diseñado por el autor.*

2.2.1. Brazo robótico con seis grados de libertad arduino stm32 robot

El brazo (6-DOF) es un tipo común de manipulador robótico que ofrece una amplia gama de movimientos y capacidades de posicionamiento.

Los grados de libertad proporciona un rango específico de movimiento. Esto puede expresarse en términos de ángulos máximos y mínimos en cada articulación.

La velocidad de movimiento del brazo robótico puede modificarse dependiendo de la aplicación y los requisitos de precisión. Puede haber diferentes velocidades para movimientos en cada grado de libertad y para movimientos cartesianos.

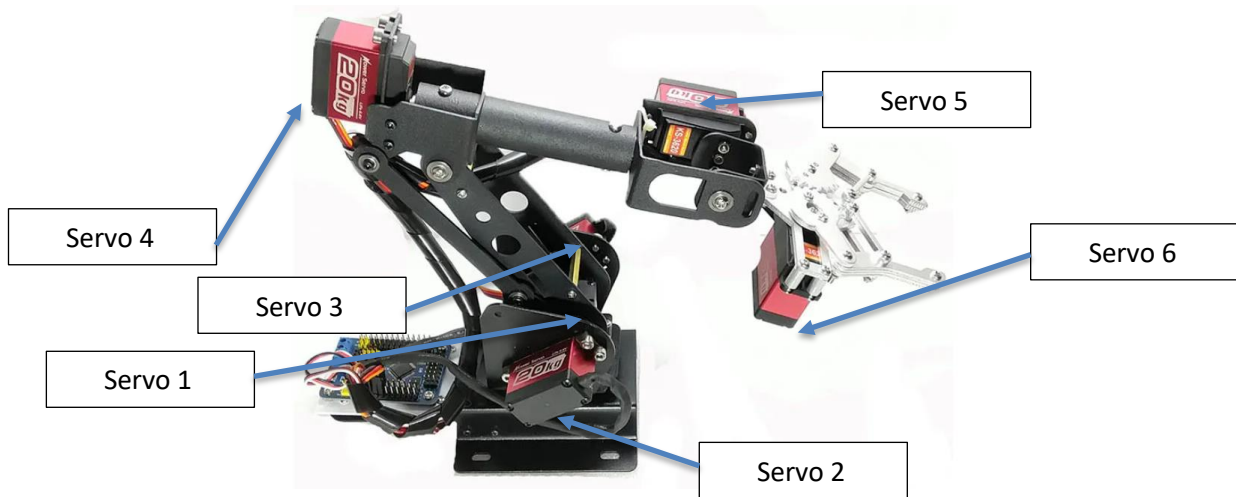
La precisión del brazo robótico para alcanzar y mantener posiciones específicas de manera consistente, puede expresarse en términos de error máximo de posicionamiento o repetibilidad.

Adaptable a cualquier controlador utilizado, los cuales pueden ser simples microcontroladores hasta controladores más complejos basados en PC o sistemas embebidos.

El software y la interfaz de programación utilizada para controlar y operar el brazo robótico, que puede variar desde interfaces gráficas de usuario (GUI) hasta lenguajes de programación específicos de robots.

Figura 3

Brazo robótico con 6 grados de libertad y definición de posiciones servos



Nota. Configuración de los servos ubicados en el brazo robótico hechos por el autor

Tabla 1

Tabla de dimensiones físicas del brazo robótico

Eje a codo	240 mm
Codo a garra	310 mm
Base a codo	123,5 mm
Codo a muñeca	145,62 mm
Brazo en reposo	120 mm
Brazo alargado	300 mm
Elongación pinza	60,7 mm

Nota. Características dimensionales del brazo robótico hechos por el autor

2.2.2. Servomotor ks3620 digital servo

Un servomotor PWM es un tipo de servomotor que se controla mediante señales PWM (modulación de ancho de pulso). Los servomotores son dispositivos electromecánicos que permiten un control preciso de la posición angular, velocidad y/o torque. Se utilizan

comúnmente en una amplia gama de aplicaciones, como robótica, modelismo, sistemas de control de posición, automatización industrial, entre otros.

El control de un servomotor PWM se realiza mediante una señal de onda cuadrada llamada señal PWM. Esta señal consiste en una serie de pulsos eléctricos de ancho variable (ancho de pulso) que se repiten a una frecuencia constante. La duración del pulso determina la posición o el ángulo deseado del servomotor.

Los servomotores PWM son capaces de girar a posiciones angulares específicas dentro de un rango determinado (generalmente 0° a 180° en servomotores estándar). La duración del pulso PWM enviado al servomotor controla la posición a la que debe moverse. Por ejemplo, un pulso de 1.5 ms puede indicar al servomotor que se mueva a 90°. ANEXO 1

Tabla 2

Datos técnicos importantes de los servos motores PWM

PRODUCTO	KS3620 DIGITAL SERVO
VOLTAJE	4.8 a 6.6 V
VELOCIDAD	0.16/60°
TORQUE	20 Kg/cm
FRECUENCIA	50 a 330 hz
ANCHO PULSO	500 – 2500 usec
PUNTO NEUTRAL	1500 usec
PESO	Alredor de 60 kg
TAMAÑO	40*40.5*20 mm

Nota. Características del servomotor PWM hechos por el autor

Figura 4

Servomotor dc de gran capacidad de torque



Nota. Modelo del servomotor Pwm, Ebay, 2020 (<https://www.ebay.com/itm/154493582978>)

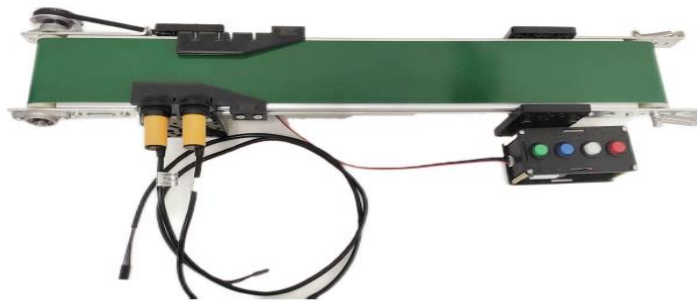
2.2.3. Banda transportadora de control de motor de paso: kit educational diy

La banda transportadora está formada por un motor de paso de hasta 12 Vdc., para que pueda ser manipulado por el usuario en el traslado de cajas en un punto determinado y además tiene incorporado en la parte superior de la banda un sensor de proximidad.

La cinta de transporte de objetos es de color verde corrugado para mejor agarre de traslado, con el objetivo de que no se caigan los objetos al suelo, finalmente la banca está fabricada de armazón de metal, para evitar que se rompa las bases de sostenimiento.

Figura 5

Banda transportadora con sus sensores de proximidad y motor de paso



Nota. Banda Transportadora Aliexpress, Younfang Smart Store, (https://www.aliexpress.us/item/3256805471479996.html?spm=a2g0o.order_list.order_list_main.63.15351802c4kKXR&gatewayAdapt=glo2usa)

2.2.4. Sensor de distancia: ir sensor switch e18-d80nk-n.

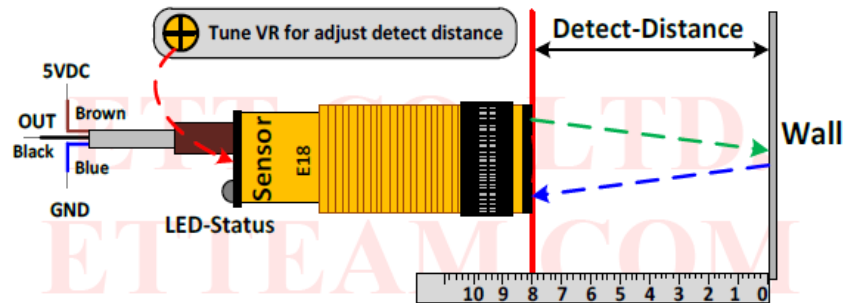
El sensor infrarrojo de distancia puede ser ajustado de 6cm hasta 80 cm, para lo cual tiene una señal lógica de salida TTL de GND (0 lógico) y 5 Vdc (1 lógico).

Se tienen cables de color que indica la configuración de alimentación y datos del sensor IR, el cual se tiene que el cable café es fuente dc, el cable azul es la tierra y el cable negro es la salida de estado detectado.

También tiene un potenciómetro, el cual servirá para modificar la detección de la distancia del objeto. (ANEXO 2)

Figura 6

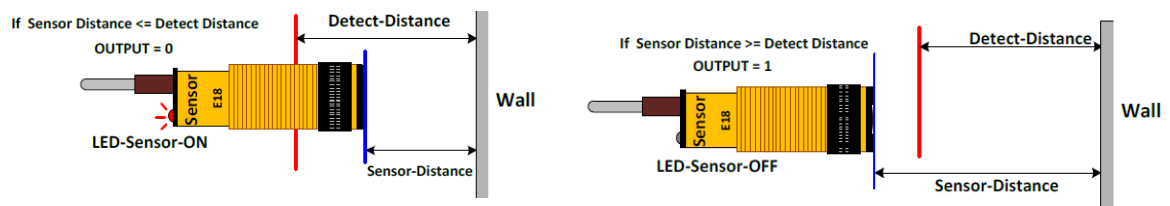
Sensor infrarrojo y sus características de seteos



El modo de funcionamiento del sensor es que cuando detecta el objeto, el dispositivo envía una señal de 0 lógico, en caso contrario sería un 1 lógico.

Figura 7

Sensor infrarrojo y sus características de seteos de aproximación



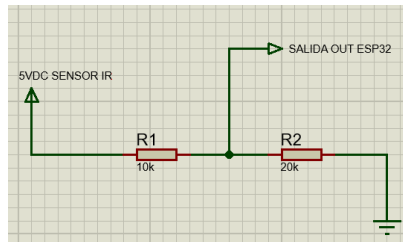
Nota. Características del sensor de proximidad infrarrojo, Twins Chip, 2023, (https://www.twinschip.com/Proximity_Sensor_E18-D80NK)

Como la señal del dispositivo envía 5 Vdc y el microcontrolador esp32 solo pueden detectar o recibir señales de 3.3 vdc, se tendría que realizar un divisor de voltaje para no dañar los canales de entrada digitales.

$$SALIDA\ OUT\ ESP32 = \frac{(R_2 + R_3)}{R_1 + R_2 + R_3} v_{5v_{DC}}$$

Figura 8

Diagrama eléctrico de conversión de señal de Vdc del sensor hasta el microcontrolador



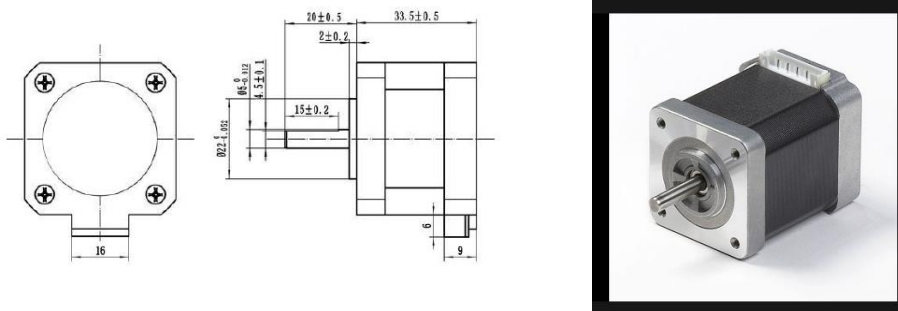
Nota. Esquemático eléctrico para variar la señal de 5Vdc a 3.3 Vdc, Proteus, propio del autor

2.2.5. Motor de paso integrado: kh42 series hybrid type, 2 phase stepping motor (1.8 degree/step) - nema 17.

El motor de paso convierte las señales de pulso en ángulo de desplazamiento. Cada vez que recibe una señal de pulso, el rotor gira un ángulo hacia adelante o atrás, es por eso, que es proporcional al número de pulsos de entradas para lo cual la velocidad es proporcional al pulso de frecuencia, por lo tanto, este motor es llamado motor bipolar. **(ANEXO 3)**

Figura 9

Diagrama de dimensiones físicas del motor de paso de la banda transportadora



Nota. Estructura del motor de paso KH42 bipolar, Elinco International JPC,2024 (<https://catalog.e-jpc.com/item/stepping-motors/stepping-motors-overall/kh4242-b95101>)

Tabla 3

Especificaciones técnicas del motor de paso

Angulo de paso	1.8°
Tasa de corriente	1.4 A

Inductancia	3.8 mH \pm 20%
Voltaje tolerancia	600 Vdc a 1 seg
Resistencia	2.3 ohm \pm 10%(20° C)
Torque de sostenimiento	0.31 N.m (44 Oz-in) Min
Peso	0.22 Kg
Número de fase	36 g- Cm2 (0.2 oz in ²)

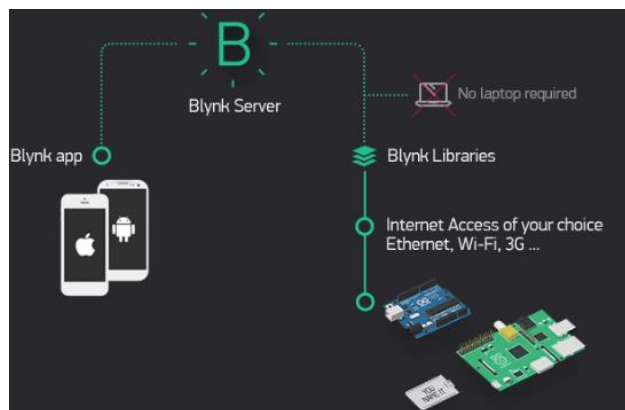
Nota. Características del motor de paso bipolar hecha por el autor

2.2.6. Comunicación del brazo robótico y banda transportadora

La comunicación del brazo robótico y banda transportadora interactivo se lo hace en modo remoto vía WIFI por protocolo de comunicación API HTTP y MQTT. Adicionalmente se relaciona con las aplicaciones IoT con base a la plataforma BLYNK para la interacción del microcontrolador ESP32 Wrover Freenove.

Figura 10

Diagrama de enlace blynk con los dispositivos



Nota. Aplicativo blynk, Tinycircuits, 2024 (<https://tinycircuits.com/blogs/learn/getting-started-with-blynk>)

2.2.7. Freenove ultimate starter kit for esp32-wrover

Freenove ESP32 es un kit de desarrollo que se basa en el microcontrolador ESP32, que es conocido por su potencia y versatilidad

Es un chip altamente integrado que combina Wi-Fi, Bluetooth de bajo consumo de energía y potentes capacidades de procesamiento en un solo chip. Ofrece un amplio rango de frecuencias de operación y es altamente flexible para una variedad de aplicaciones.

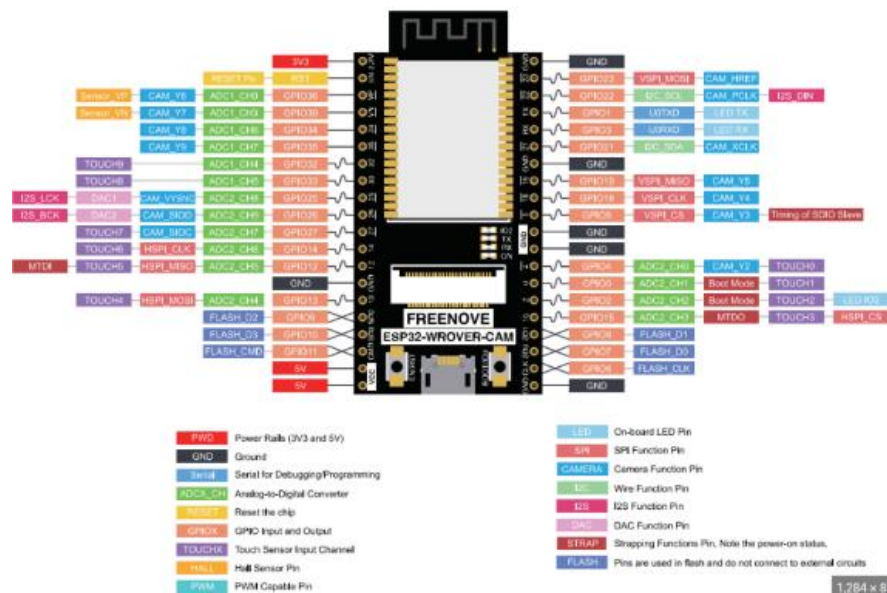
Freenove ofrece una placa de desarrollo que incluye el ESP32 y varios componentes adicionales para facilitar el desarrollo de proyectos. Esto puede incluir puertos de entrada/salida (GPIO), interfaces de comunicación como UART, SPI, I2C, así como puertos USB para programación y alimentación, adicionalmente, integra módulos Wi-Fi y Bluetooth, lo que permite la conectividad inalámbrica a redes locales y dispositivos periféricos compatibles con Bluetooth.

Los pines GPIO permiten la conexión de sensores, actuadores y otros dispositivos externos al microcontrolador para interactuar con el entorno, además es compatible con el entorno de programación Arduino, lo que significa que puedes programarlo utilizando el lenguaje de programación C y aprovechar una amplia gama de bibliotecas y ejemplos disponibles.

Freenove generalmente proporciona una amplia documentación y tutoriales para ayudar a los usuarios a comenzar y desarrollar proyectos con el kit ESP32. Esto puede incluir manuales de usuario, guías de inicio rápido y ejemplos de código. **(ANEXO 4)**

Figura 11

Esquema diseño ESP32



Note. Diagrama de pines y configuraciones del microcontrolador ESP32, Ramdom Nerd Tutorial, 2013 (<https://randomnerdtutorials.com/getting-started-freenove-esp32-wrover-cam/>)

2.2.8. A4988 stepper motor driver board

El A4988 es un controlador de corriente constante que puede manejar motores paso a paso bipolares o unipolares. Está diseñado para controlar la velocidad y la dirección de los motores paso a paso en aplicaciones donde se requiere un movimiento preciso y controlado.

Permite ajustar la corriente del motor mediante un potenciómetro en la placa, lo que permite adaptar la corriente a las necesidades específicas del motor y de la aplicación.

Se utiliza en una amplia variedad de aplicaciones que requieren control preciso de motores paso a paso, como impresoras 3D, máquinas CNC, sistemas de posicionamiento, robótica, cámaras de seguimiento, entre otros. **(ANEXO 5)**

Figura 12

Diseño del driver de potencia A4988



Nota. Modelo driver de potencia para motor de paso, 3dshop.uk, 2024 (<https://3dshop.uk/shop/4988-stepper-driver-module-and-heat-sink/>)

Tabla 4

Datos técnicos del dispositivo integrado de potencia

Modelo	A4988
Color	Verde o Rojo
Máxima Corriente	2 A
Tensión Máxima	35 V
Microsteps	16
Rs típico	0.05, 0.1 o 0.2

Nota. Datos Técnicos del driver de potencia para motor de paso diseñado por el autor

2.2.9. Aplicaciones del software blynk

Blynk es una plataforma de desarrollo IoT (Internet de las cosas), el cual permite a los usuarios controlar dispositivos electrónicos de manera remota a través de una aplicación móvil.

Proporciona una aplicación móvil para dispositivos iOS y Android que actúa como la interfaz de usuario para controlar y monitorear dispositivos conectados. La aplicación ofrece una interfaz gráfica de manera que, los usuarios puedan crear paneles personalizados con botones, deslizadores, gráficos y otros widgets para interactuar con sus dispositivos.

Ofrece una plataforma en la nube que actúa como intermediario entre la aplicación móvil y los dispositivos conectados. La plataforma proporciona servicios de autenticación, gestión de usuarios y comunicación segura entre la aplicación y los dispositivos.

Blynk proporciona librerías de desarrollo para una variedad de plataformas de hardware populares, incluyendo Arduino, Raspberry Pi, ESP8266, ESP32 y muchas más. Estas librerías facilitan la conexión de dispositivos a la plataforma Blynk y la comunicación con la aplicación móvil.

La aplicación Blynk ofrece una amplia gama de widgets personalizables que los usuarios pueden agregar a sus paneles para controlar y monitorear sus dispositivos. Estos widgets incluyen botones, deslizadores, gráficos, medidores, notificaciones push y más.

Figura 13

Dispositivo celular usando plataforma Blynk



Nota. Creación de Widget. BLYNK,2024 (<https://ny3.blynk.cloud/dashboard/149686/templates/edit/381126/datastreams>)

Es compatible con una variedad de protocolos de comunicación, incluyendo Wi-Fi, Bluetooth, Ethernet y GSM, la cual permite a los usuarios conectar una amplia variedad de dispositivos a la plataforma.

Utiliza medidas de seguridad robustas para proteger la comunicación entre la aplicación móvil y los dispositivos conectados. Lo que permitirá la autenticación de usuarios.

Figura 14

Modelo de creación de control con dispositivo celular para automatización IoT



Nota. Ejemplo de automatización IoT, *hackster.io*, 2024 (<https://www.hackster.io/embeddedlab786/iot-and-manual-switch-control-home-appliances-using-blynk-7d2236>)

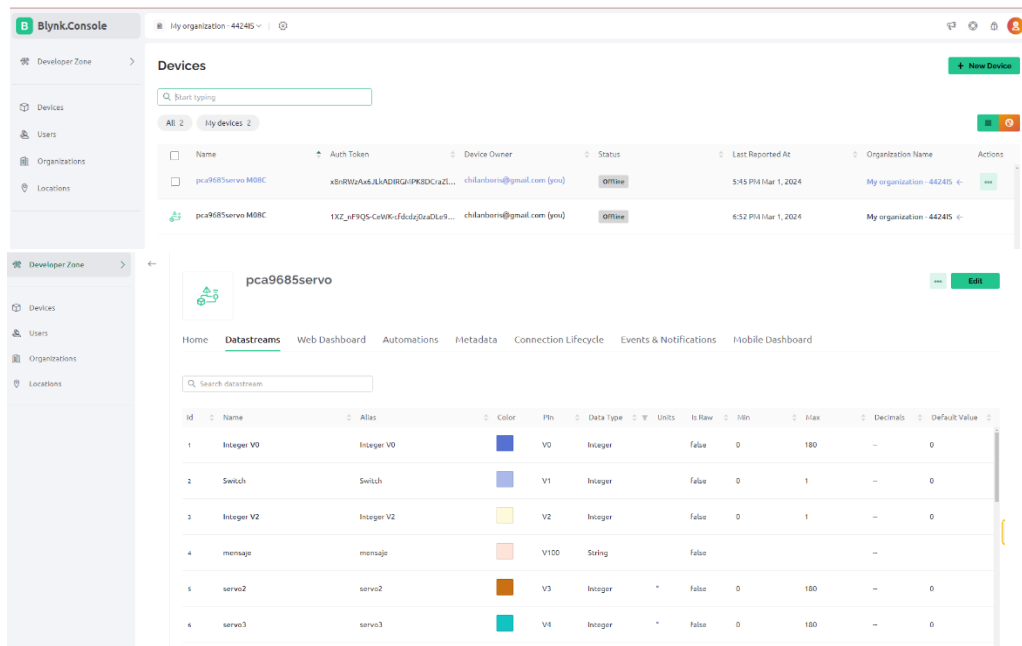
2.2.10. Plataforma blynk console.

Para poder crear y organizar las variables del proyecto, se debe acceder a la plataforma BLYNK en donde deberá crear un usuario para poder acceder a las herramientas de programación, el cual se tendrá que crear un Template Device para que pueda guardar las variables de automatización (integer, string, discrete), estos son denominados Virtual Pin.

Para que pueda ser integrado al proyecto, deberá bajar la aplicación BLYNK APP e instalarlo en el celular donde elegirá los widgets y la comunicación con el microcontrolador.

Tabla 5

Plataforma de variables usadas en el proyecto en la plataforma Blynk



Nota. Consola de variables en la plataforma BLYNK IoT, BLYNK. 2024 (<https://blynk.io/blynk-iot-low-code-software-platform>)

2.2.11. Software de programación arduino ide

Arduino se destaca por su facilidad de uso, por lo cual, es una opción para principiantes en electrónica y programación. El ambiente de desarrollo integrado (IDE), es simple y amigable, con una interfaz intuitiva que permite a los usuarios escribir, cargar y depurar código de manera sencilla.

Utiliza hardware de código abierto, lo que significa que las especificaciones de sus placas están disponibles para que cualquiera las utilice y modifique según sus necesidades. Esto permite toda una amplia gama de placas Arduino compatibles y variantes personalizadas para adaptarse a diferentes aplicaciones.

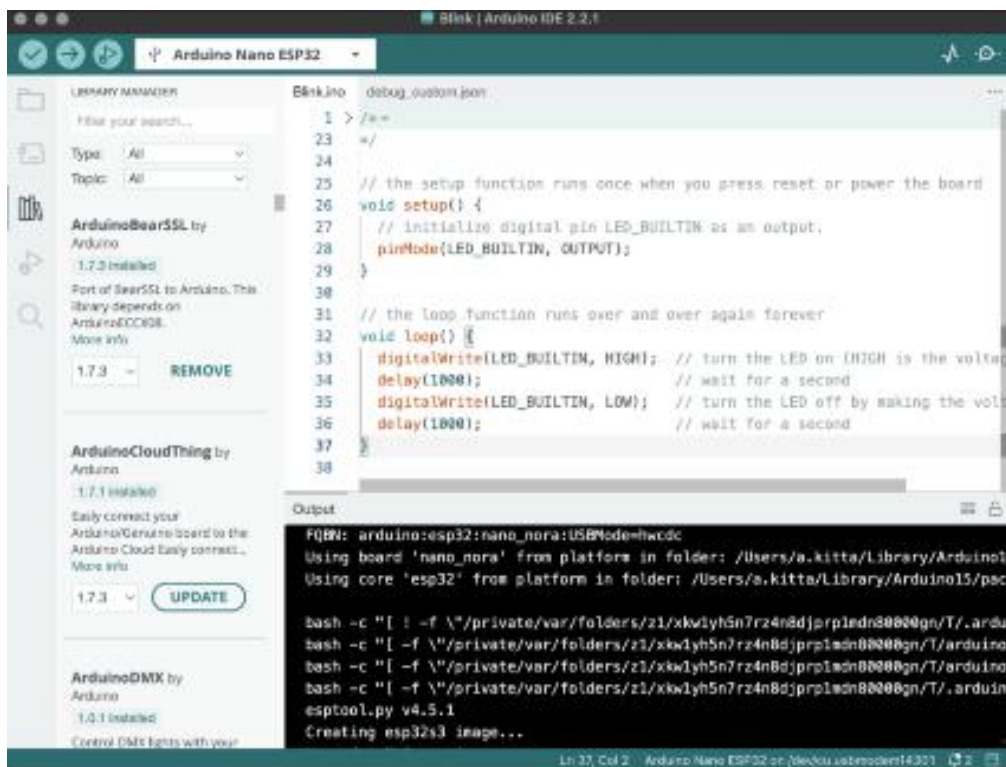
Arduino ofrece una variedad de placas con diferentes características y especificaciones para adaptarse a diversas necesidades y proyectos. Desde placas básicas como Arduino Uno hasta opciones más avanzadas como Arduino Mega o Arduino Due, hay una placa Arduino para casi cualquier aplicación.

La comunidad de Arduino es grande y activa, con recursos en línea suficientes, incluyendo tutoriales, documentación, foros de discusión y proyectos compartidos. Esto proporciona un valioso soporte para los usuarios de Arduino, independientemente de su nivel de experiencia.

La amplia variedad de periféricos y sensores permite a los usuarios desarrollar una amplia gama de proyectos interactivos y sistemas embebidos. Esta flexibilidad es una de las razones por las que Arduino es tan popular en la creación de prototipos en la educación.

Figura 15

Plataforma de programación arduino



Nota. Plataforma Arduino IDE v2.0, Flathub,2022 (<https://flathub.org/apps/cc.arduino.IDE2>)

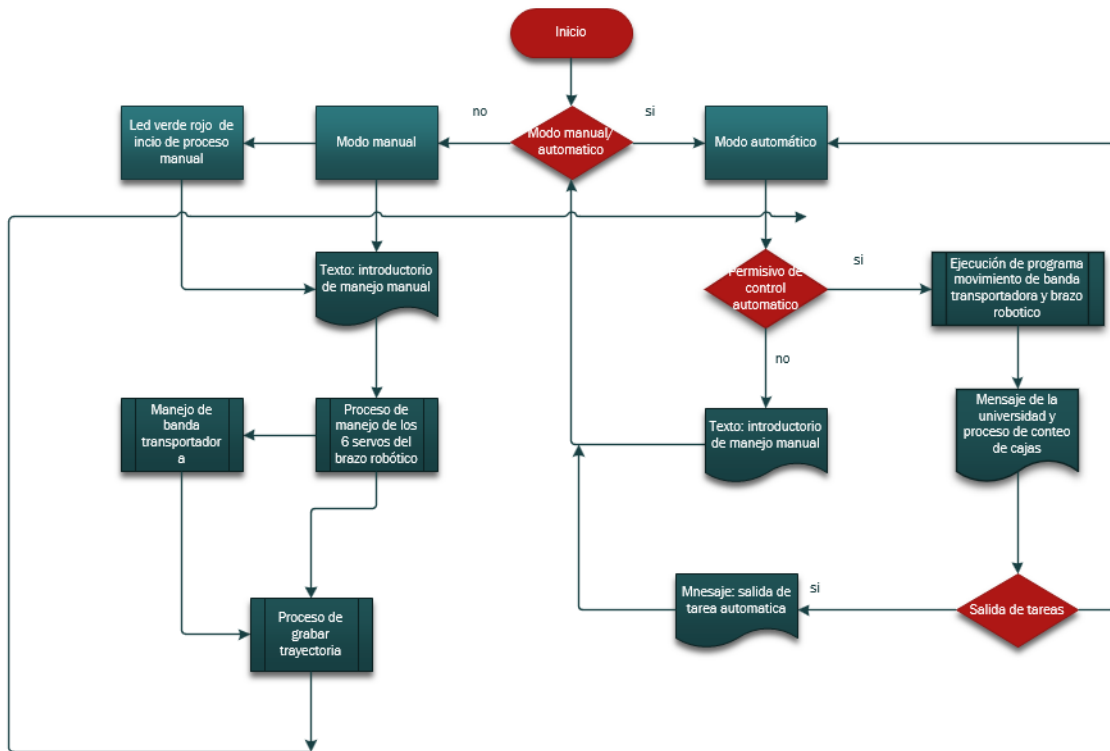
2.3 Metodología de trabajo.

2.3.1. Estructura general (Anexo 1)

El diagrama de flujo representa las tareas de control manual y automático; los subprocesos detallados en el mismo anexo para operaciones individuales de brazo y banda transportadora. (ANEXO 6 Y 7)

Figura 16

Diagrama de flujo de tarea del brazo robótico y banda transportadora



Nota. Visión general de diagrama de flujo del proyecto de control de brazo y banda transportadora diseñado por el autor

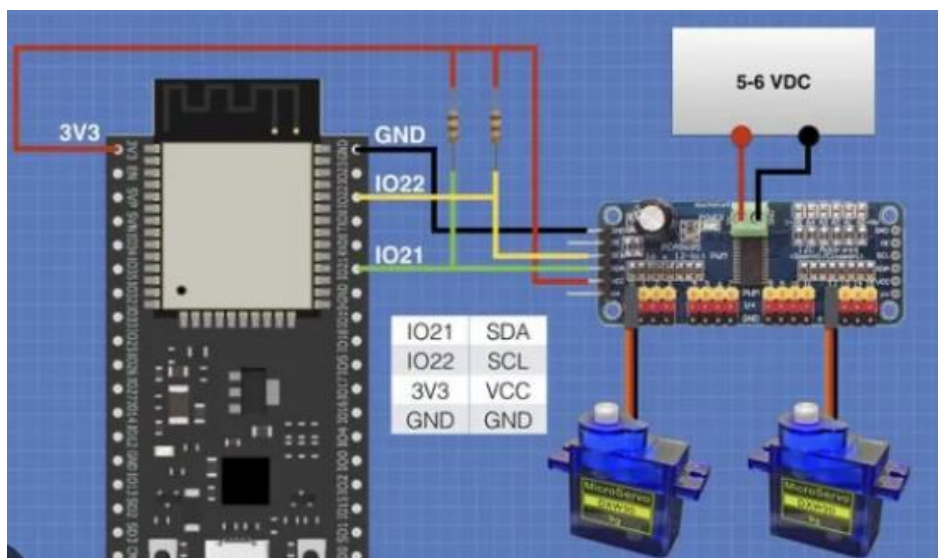
2.3.2. Configuración y programación PWM del controlador con el dispositivo PCA9685 para control del brazo robótico.

2.3.2.1. Conexión hardware

La comunicación serial entre el dispositivo PCA9685 con el microcontrolador ESP32 Worover, es por el protocolo de comunicación I2c en los pines establecidos en 21 y 22 GPIO, para la cual se le ha asignado la dirección address de 0x60 recomendado por la placa driver de potencia PWM.

Figura 17

Esquema de conexiones de los dispositivos con el driver de potencia y servos



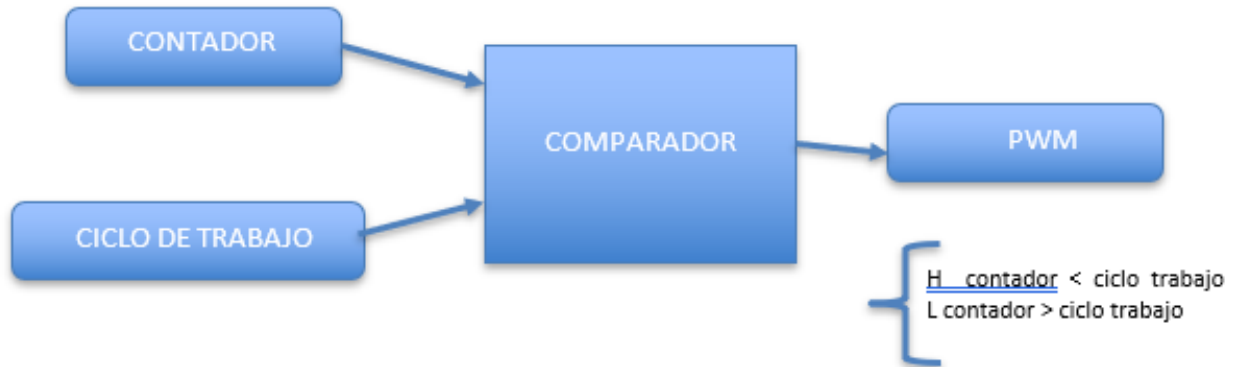
Nota. *Conexión del controlador y la tarjeta de potencia PCA9685 con comunicación I2c diseñado por el autor*

2.3.2.2. Teoría de programación PWM.

Para poder entender la programación en software de un proceso de señal PWM, se debe observar el diagrama lógico siguiente.

Figura 18

Lógica de programación PWM

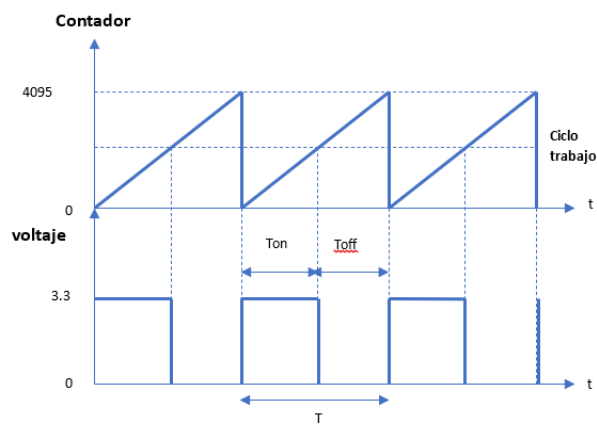


Nota. Teoría proceso brazo servo PWM diseñado por el autor

El controlador generará un contador interno para este caso sería una resolución de 12 bits (0 hasta 4095) y se comparará con un ciclo de trabajo para tener referencia de salida, normalmente si el contador es menor que el ciclo de trabajo, se generará un alto en caso contrario se limitará a cero. Esta lógica de software es comparable al circuito electrónico denominado diente de sierra con un contador integrado 555.

Figura 19

Lógica de señales PWM



Nota. Teoría de señales brazo servo diseñado por el autor

Primero se tendrá que configurar las especificaciones técnicas de los servos motores con las características de los parámetros, los cuales el dispositivo tendrá la dirección de comunicación 0x60 y que el ancho de pulso mínimo es de 500 y máximo es de 2500 con una frecuencia total de 50 Hz. **(ANEXO 6 Y 7)**

```
Adafruit_PWMServoDriver pwmT0 = Adafruit_PWMServoDriver(0x60);
#define MIN_PULSE_WIDTH 500
#define MAX_PULSE_WIDTH 2500
#define FREQUENCY 50
```

La función para hacer las conversiones de los ángulos provenientes de la señal análoga y también elegir cual servo es necesario que se ejecute, es necesario usar la función MAP(), esto permitirá hacer una transformación de señal desde 0 hasta 180 con un rango de ancho de pulso mínimo y máximo.

```
void setServo(uint8_t n_servo, int angulo) {
    int duty;
    duty=map(angulo,0,180,MIN_PULSE_WIDTH, MAX_PULSE_WIDTH);
    int valor_analogo = int(float(duty) / 1000000 * FREQUENCY * 4096);
    pwmT0.setPWM(n_servo, 0, valor_analogo);
}
```

```
setServo(0, servo1);
setServo(1, servo2);
setServo(2, servo3);
setServo(3, servo4);
setServo(4, servo5);
setServo(5, servo6);
```

Código de trayectoria de brazo robótico, para poder ejecutar las posiciones guardas en una matriz de información.

```
if(flagCaja==1){
    Blynk.virtualWrite(V101,"ejecutando\n");
    Blynk.virtualWrite(V101,"CAJA:",cntCaja,"\n");
    for(trayectoria=1;trayectoria<=cntPos;trayectoria++)
        for( cnt_servos=1; cnt_servos<=3;cnt_servos++){//contadores de servos del
    if(posiciones[trayectoria][cnt_servos]<posiciones[trayectoria+1][cnt_servos]){
        for(tempAngulo=posiciones[trayectoria][cnt_servos];
tempAngulo<=posiciones[trayectoria+1][cnt_servos];tempAngulo++){ //recorrido del angulo
            anchopulso = map(tempAngulo, 0, 180, MIN_PULSE_WIDTH, MAX_PULSE_WIDTH);
            tmpAnalogValue = int(float(anchopulso) / 1000000 * FREQUENCY * 4096);
            //base[cnt_servos-1]=analogue_value;
            pwmT0.setPWM(cnt_servos-1,0,tmpAnalogValue);
            //Blynk.virtualWrite(V101,analog_value,"\n");
```



```

        delay(delayServos);
        while(buttonStop==1){BlynkEdgent.run();} //bucle de emerg
        BlynkEdgent.run();
    } //cierre del recorrido del angulo
} //cierre de logica de trayectoria positiva
else
if(posiciones[trayectoria][cnt_servos]>posiciones[trayectoria+1][cnt_servos]){ //trayectoria
    for( tempAngulo=posiciones[trayectoria][cnt_servos];
tempAngulo>=posiciones[trayectoria+1][cnt_servos];tempAngulo--){ //recorrido del angulo
        anchopulso = map(tempAngulo, 0, 180, MIN_PULSE_WIDTH, MAX_PULSE_WIDTH);
        tmpAnalogValue = int(float(anchopulso) / 1000000 * FREQUENCY * 4096);
        //base[cnt_servos-1]=analogue_value;
        pwmT0.setPWM(cnt_servos-1,0,tmpAnalogValue);
        delay(delayServos);
        while(buttonStop==1){
            BlynkEdgent.run();
        } //bucle de emergencia
        BlynkEdgent.run();
    } //cierre del recorrido del angulo
} //cierre de trayectoria negativa
else{//trayectorias iguales
    tempAngulo=posiciones[trayectoria][cnt_servos];
    anchopulso = map(tempAngulo, 0, 180, MIN_PULSE_WIDTH, MAX_PULSE_WIDTH);
    tmpAnalogValue = int(float(anchopulso) / 1000000 * FREQUENCY * 4096);
    //base[cnt_servos-1]=analogue_value;
    pwmT0.setPWM(cnt_servos1,0,tmpAnalogValue);
} //cierre de trayectorias iguales
} //cierre de contadores del servo del motor
while(digitalRead(detecBox)==0){}
//vTaskDelay(2);
} //caja encontrada

```

Tabla 6

Tabla de movimientos físicos de los servo motores

SERVO 1 (base)	0° - 180°
SERVO 2 (codo)	0° - 110°
SERVO 3 (giro codo)	0° - 82° (PUNTO MUERTO) - 180°
SERVO 4 (antebrazo)	110° - 136°
SERVO 5 (muñeca)	0° - 85° (PUNTO MUERTO) - 180°
SERVO 6 (gripper)	0° - 88°

Figura 20

Montaje físico y conexiones del brazo robótico



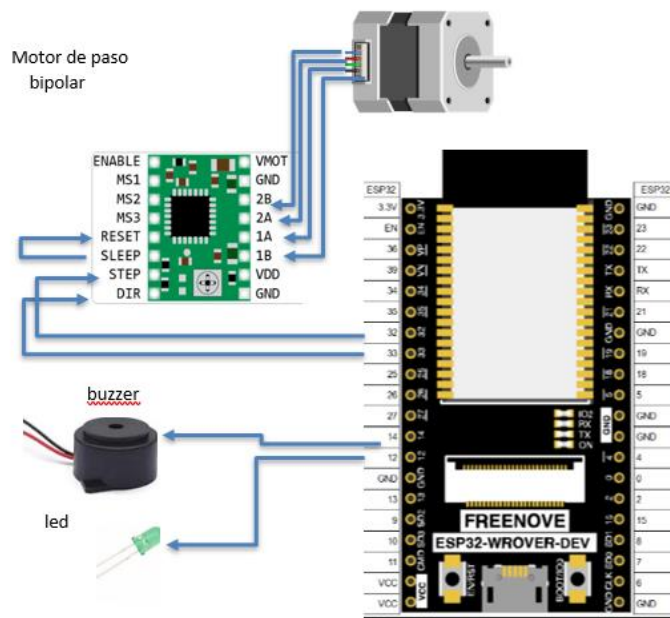
Nota. *Instalación y conexiones del brazo servo PWM PCA9685 diseñado por el autor*

2.3.3. Configuración y programación del dispositivo A4988 para control de banda transportadora.

2.3.3.1. Conexión hardware

Figura 21

Esquema de conexiones con la banda transportadora



Nota. *Diagrama de conexionado con la banda transportadora y el microcontrolador diseñado por el autor*

Para poder proteger el driver de potencia contra los picos de corrientes del motor de paso, se deberá establecer los parámetros de operación segura.

El límite corriente que se recomienda es de 1A, la resistencia $R_{cs}=0.068$ por lo que se deberá aplicar la fórmula:

$$I_{max} = \frac{V_{ref}}{8 R_{cs}}$$

Se tendría el valor de voltaje referencia de 540 mV para protección de corriente inrush, además se lo configuró como motor de paso completo para evitar sobre corrientes.

Tabla 7

Tabla de configuración de pines para movimiento del motor de paso

MS1	MS2	MS3	Resolución de micropasos
Bajo	Bajo	Bajo	Paso completo
Alta	Bajo	Bajo	1/2 paso
Bajo	Alta	Bajo	1/4 de paso
Alta	Alta	Bajo	1/8 paso
Alta	Alta	Alta	Paso 1/16

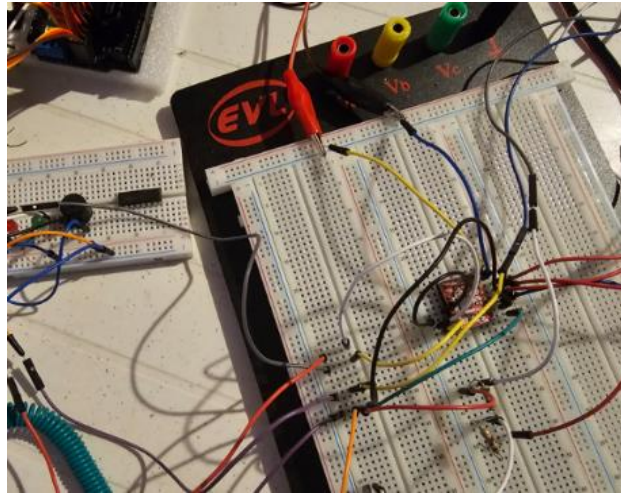
Nota. *Tabla de configuración de pines para movimientos del motor de paso hechos por el autor*

La lógica de programación se lo realizó por las salidas en los pines 33 (dir) y 32 (step) del microcontrolador ESP32, el cual el control se encargará cambiando la dirección y los pasos de movimiento del driver de potencia.

```
const int dirPin =33;
const int stepPin = 32;
```

Figura 22

Conexión del driver de potencia A4988 para el control de motor de paso



Nota. Instalación y conexiones del driver A4988 diseñado por el autor

2.3.4 Programacion blynk

Blynk es una plataforma que permite crear aplicaciones para dispositivos IoT (Internet de las cosas) de una manera sencilla y rápida.

Paso 1: Descargar la aplicación Blynk

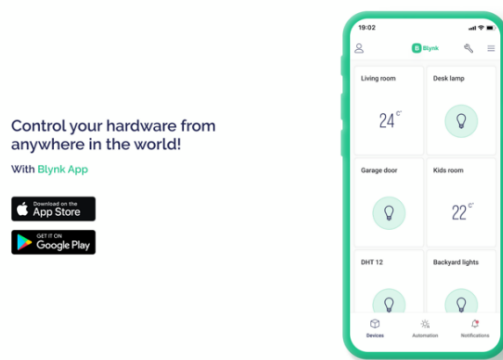
Abrir la tienda de aplicaciones en el dispositivo móvil (App Store para iOS o Google Play Store para Android).

Buscar "Blynk" en la tienda.

Descargar e instalar la aplicación Blynk en el dispositivo.

Figura 23

APP Blynk para dispositivos android



Nota. App Blynk mobil aps. 2024 (<https://docs.blynk.io/en/downloads/blynk-apps-for-ios-and-android>)

Paso 2: Crear una cuenta en Blynk

Abre la aplicación Blynk en tu dispositivo móvil.

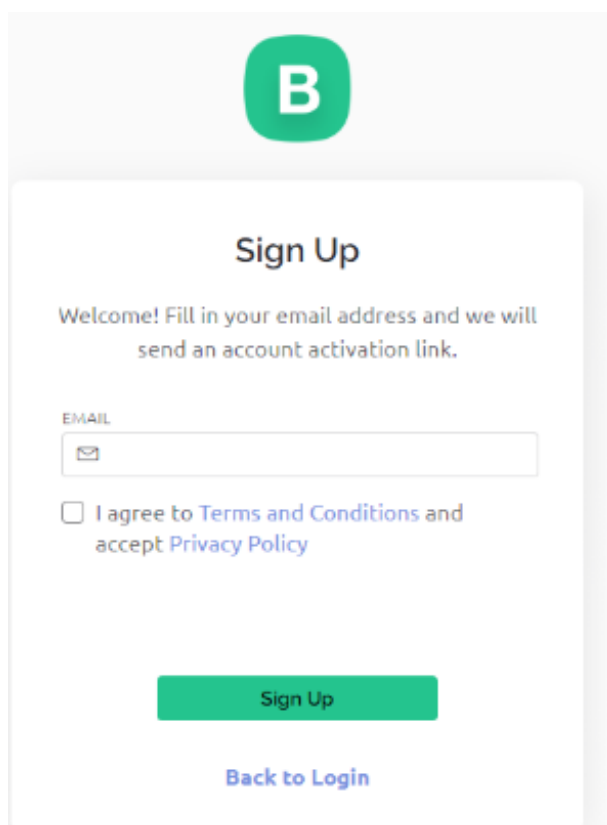
Toca en "Crear una cuenta" para registrarte.

Ingresa la dirección de correo electrónico y crear una contraseña.

Confirmar la dirección de correo electrónico siguiendo las instrucciones que se recibirá por correo electrónico.

Figura 24

Creación de usuario en la página de Blynk



Nota. Creación de usuario. BLYNK,2024 (<https://ny3.blynk.cloud/dashboard/149686/templates/edit/381126/datastreams>)

Paso 3: Crear un nuevo proyecto en Blynk

Una vez iniciada la sesión, tocar en el botón "+" para crear un nuevo proyecto.

Asigna un nombre al proyecto y selecciona el dispositivo con el que trabajarás (para este caso es el ESP32).

Paso 4: Obtener el token de autenticación

Después de crear el proyecto, recibirás un correo electrónico con el token de autenticación. Este token es necesario para conectar tu dispositivo con la aplicación Blynk.

Copiar el token de autenticación desde el correo electrónico o desde la aplicación Blynk.

Paso 5: Configurar el hardware y la aplicación Blynk

Conectar el dispositivo IoT (Arduino, Raspberry Pi, etc.) a Internet y asegúrate de tener instalada la biblioteca o el firmware necesario para Blynk.

Abrir el código de ejemplo de Blynk para el dispositivo en el IDE correspondiente (Arduino IDE, Raspberry Pi, etc.).

Pegar el token de autenticación que copiaste anteriormente en el código.

Sube el código a tu dispositivo.

Paso 6: Crear widgets en la aplicación Blynk

Regresar a la aplicación Blynk en tu dispositivo móvil.

Tocar en el proyecto que creaste anteriormente.

Tocar en el botón "+" para agregar widgets como botones, deslizadores, gráficos, etc.

Configurar cada widget con las propiedades deseadas (pin, modo, intervalo, etc.).

Paso 7: Probar la conexión

Una vez que hayas configurado los widgets, asegurarse de que el dispositivo IoT esté encendido y conectado a Internet.

Tocar en el botón de reproducción en la esquina superior derecha de la aplicación Blynk para iniciar la aplicación.

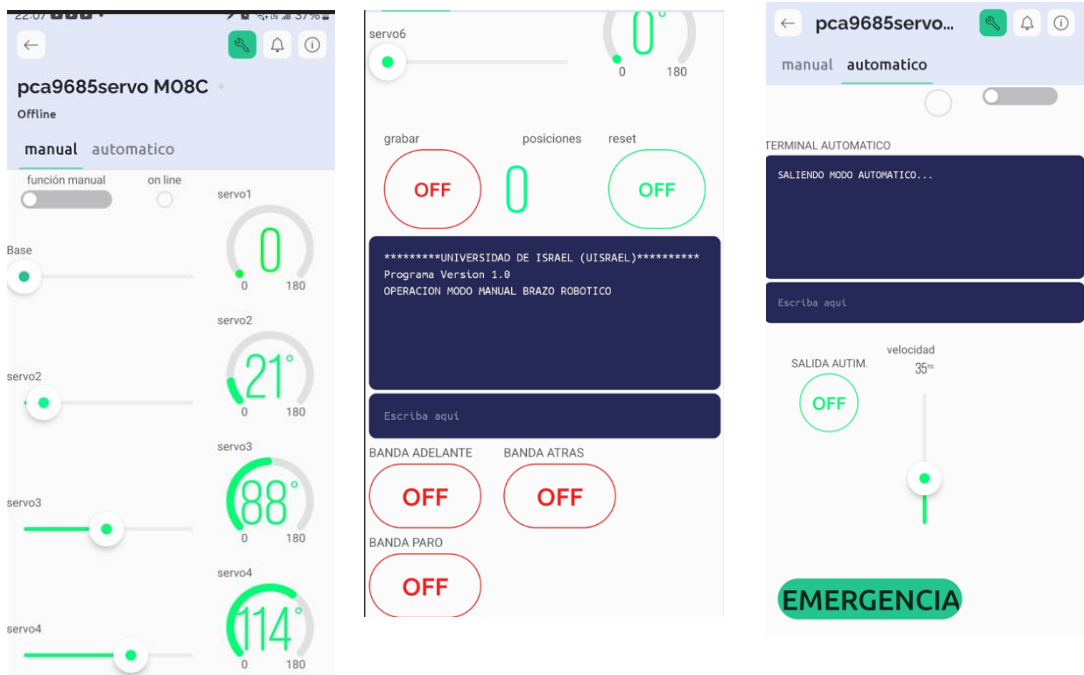
Tabla 8*Tabla de variables Blynk:*

<input type="checkbox"/>	Id	Name	Alias	Color	Pin	Data Type	Units	Is Raw	Min	Max
	1	Integer V0	Integer V0		V0	Integer		false	0	180
	2	Switch	Switch		V1	Integer		false	0	1
	3	Integer V2	Integer V2		V2	Integer		false	0	1
	4	mensaje	mensaje		V100	String		false		
	5	servo2	servo2		V3	Integer	°	false	0	180
	6	servo3	servo3		V4	Integer	°	false	0	180
	7	servo4	servo4		V5	Integer	°	false	0	180
	8	servo5	servo5		V6	Integer	°	false	0	180
	9	servo6	servo6		V7	Integer	°	false	0	180
	10	grabar	grabar		V8	Integer		false	0	1
	11	posiciones	posiciones		V9	Integer		false	0	20
	12	reset	reset		V10	Integer		false	0	1
	13	terminal2	terminal2		V101	String		false		
	14	SALIR_AUTOMATICO	SALIR AUTOMATICO		V11	Integer		false	0	1
	15	delay_servos	delay servos		V12	Integer	ms	false	0	100
	16	EMERGENCIA	EMERGENCIA		V13	Integer		false	0	1
	17	ADELANTE	ADELANTE		V14	Integer		false	0	1
	18	ATRAS	ATRAS		V15	Integer		false	0	1
	19	PARO	PARO		V16	Integer		false	0	1

Nota. *Tabla de variables de datastream en plataforma BLYNK diseñado por el autor*

Figura 25

Plataforma de imagen en el teléfono celular.



Nota. Diseño de control para teléfono móvil diseñado por el autor

2.4. Validación de la propuesta.

Se ha elegido el personal con experiencia en comunicaciones de redes, años de experiencia laboral y proyectos relacionado al área de la automatización dentro de la empresa Hidroeléctrica Marcel Laniado de Wind y parte de la prestigiosa Universidad de Israel en el área de la enseñanza educativa.

En la tabla se da un resumen breve de los profesionales escogidos.

Tabla 9

Perfil de validadores

Nombres y Apellidos	Años de experiencia	Titulación Académica	Cargo
Miguel Cedeño	15	Ingeniero en Telecomunicaciones	Supervisor SCADA Central Hidroeléctrica Marcel Laniado de Wind
Roberto Pérez Suarez	10	Ingeniero en Potencia y Control	Supervisor Eléctrico de Automatización Central Hidroeléctrica Marcel Laniado de Wind
Maryory Urdaneta Herrera	20	PHD en Ingeniería Eléctrica	Profesor de Maestrías e Investigación en la Universidad de Israel

Tabla 10*Criterios de evaluación*

Criterios	Descripción
Impacto	Representa el alcance que tendrá el modelo de gestión y su representatividad en la generación de valor público.
Aplicabilidad	La capacidad de implementación del modelo considerando que los contenidos de la propuesta sean aplicables
Conceptualización	Los componentes de la propuesta tienen como base conceptos y teorías propias de la gestión por resultados de manera sistémica y articulada.
Actualidad	Los contenidos de la propuesta consideran los procedimientos actuales y los cambios científicos y tecnológicos que se producen en la nueva gestión pública.
Calidad Técnica	Miden los atributos cualitativos del contenido de la propuesta.
Factibilidad	Nivel de utilización del modelo propuesto por parte de la Entidad.
Pertinencia	Los contenidos de la propuesta son conducentes, concernientes y convenientes para solucionar el problema planteado.

Tabla 11*Validador 1*

Nombres y Apellidos	Años de experiencia	Titulación Académica	Cargo
Miguel Cedeño	13	Magister Automatización y control e Ingeniero en Sistemas y Telecomunicaciones.	Supervisor SCADA Central Hidroeléctrica Marcel Laniado de Wind

Tabla 12*Escala de evaluación por Mgtr. Miguel Cedeño*

CRITERIOS	EVALUACION SEGUN IMPORTANCIA Y REPRESENTATIVIDAD				
	En Total Desacuerdo	En Desacuerdo	Ni de Acuerdo Ni en Desacuerdo	De Acuerdo	Totalmente Acuerdo
Impacto					X
Aplicabilidad					X
Conceptualización					X
Actualidad					X
Calidad Técnica					X
Factibilidad					X
Pertinencia					X

Firma:

Tabla 13*Validador 2*

Nombres y Apellidos	Años de experiencia	Titulación Académica	Cargo
Roberto Pérez Suarez	10	Magister Automatización y control e Ingeniero en Sistemas de Potencia	Supervisor Eléctrico de Automatización Central Hidroeléctrica Marcel Laniado de Wind

Tabla 14*Escala de evaluación por Mgtr. Roberto Pérez Suarez*

CRITERIOS	EVALUACION SEGUN IMPORTANCIA Y REPRESENTATIVIDAD				
	En Total Desacuerdo	En Desacuerdo	Ni de Acuerdo Ni en Desacuerdo	De Acuerdo	Totalmente Acuerdo
Impacto					X
Aplicabilidad					X
Conceptualización					X
Actualidad					X
Calidad Técnica					X
Factibilidad					X
Pertinencia					X

Firma:

Tabla 15*Validador 3*

Nombres y Apellidos	Años de experiencia	Titulación Académica	Cargo
Maryory Urdaneta Herrera	20	PHD en Ingeniería Eléctrica	Profesor de Maestrías e Investigación en la Universidad de Israel

Tabla 16*Escala de evaluación por PHD. Maryory Urdaneta Herrera*

CRITERIOS	EVALUACION SEGUN IMPORTANCIA Y REPRESENTATIVIDAD				
	En Total Desacuerdo	En Desacuerdo	Ni de Acuerdo Ni en Desacuerdo	De Acuerdo	Totalmente Acuerdo
Impacto					X
Aplicabilidad					X
Conceptualización					X
Actualidad					X
Calidad Técnica					X
Factibilidad					X
Pertinencia					X

Firma:

2.5. Matriz de articulación de la propuesta

Se resume la matriz de la articulación del producto en la cual se basó sustentos teóricos, metodológicos y datos tecnológicos.

Tabla 17

Matriz de articulación

Ejes o partes principales del proyecto	Breve descripción de los resultados de cada parte	Sustento teórico que se aplicó en la construcción del proyecto	Metodologías, herramientas técnicas y tecnológicas que se emplearon
<p>1</p> <p>Definición: Elementos electrónicos, de control de aplicación de programación, de comunicaciones, variables de entrada a monitorear, variables de salida a controlar, etc.</p>	<p>Manejo y funcionamiento de configuración de brazos robóticos y bandas transportadoras, sensores de presencia infrarrojo, protección de corriente inrush y manejo de servos motores PWM.</p>	<p>Inteligencia Artificial</p> <p>Internet de las Cosas</p> <p>Matemáticas Aplicadas</p> <p>Comunicaciones Inalámbricas</p>	<p>Se comenzó con la teoría de implementar una herramienta de plataforma IoT como BLYNK para poder demostrar la facilidad de enseñanza de manejo de robótica y crear una maqueta de una industria 4.0</p>
<p>2</p> <p>Diseño: Circuito electrónico, de control, compatibilidad de componentes y aplicaciones, programación.</p>	<p>Debido al uso de la plataforma BLYNK IoT y sus variables virtuales, se basó en un punto de programación de microcontroladores de Arduino IDE v2.0</p>	<p>Se basó principalmente en el lenguaje C para microcontroladores en la plataforma Arduino IDE v2.0 para poder implementar programación PWM, control de Motor de Paso y aplicativo software para comunicación BLYNK Cloud IoT con las variables virtuales creadas en el teléfono móvil.</p>	<p>La configuración en el celular combinado con la programación en plataforma de microcontroladores hacen más fácil poder integrar una</p>

			.	aplicación en el teléfono móvil.
3	Implementación: Aplicaciones IoT que se basa en la plataforma BLYNK, comunicación WIFI, uso de enlaces CLOUD, señales de entrada y salida para control de los elementos driver de potencia de los servos motores PWM y motor de paso bipolar	Adaptación de señales wifi junto con los elementos de comunicación de librerías BLYNK, permiten que el control de los equipos sean estables con las debidos cuidados de señales de 3.3 Vdc, 5 Vdc y 9 Vdc, los cuales permiten el funcionamiento correcto de los motores instalados en cada uno de los elementos de control.	Protocolos de comunicación Tcp / IP	Equipos de motores y señalización sin problema con el enlace de comunicación en la nube (cloud)

Análisis de resultados. Presentación y discusión.

En La funcionalidad del brazo robótico, se ha tenido precisión en la operabilidad de control y además la exactitud cuando se realiza la tarea asignada cumpliendo con el algoritmo de trayectoria desarrollada y basada con teorías iniciales de manejo de brazos robóticos.

La velocidad de operación es en forma manual, por lo que, al desarrollar la función establecida no afectará las trayectorias grabadas.

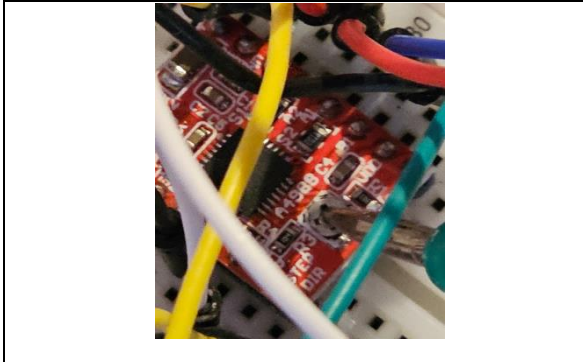
Con respecto a la operación de la banda transportadora, se pudo verificar la fluidez y Continuidad del movimiento de la banda transportadora, además de la capacidad de carga que no se observó ningún comportamiento anormal en su operación de transporte de cajas.

En la Integración con la Plataforma Blynk, se verificó que la capacidad para controlar el brazo robótico y la banda transportadora de forma remota a través de la plataforma Blynk, se lo realizó en forma satisfactoria, como dato adicional, se observó que el monitoreo en forma de tiempo real se realizó sin ningún inconveniente o problema de comunicación.

2.6. Evidencia de Instalación de equipos:

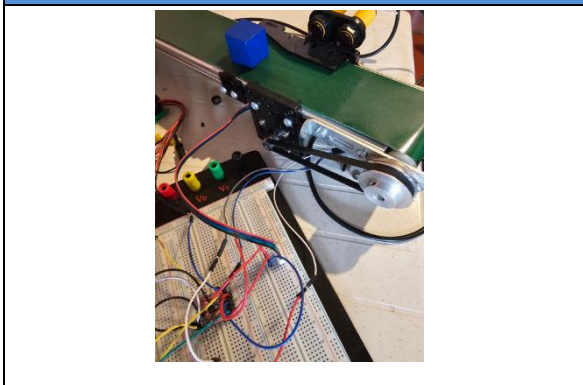
<p>Pruebas de señal PWM para motores servo al 5.10% (modo inicial)</p>	<p>Pruebas de señal PWM para motores servo al 98%</p>
<p>Cableado del driver de potencia PWM PCA9685</p>	<p>Prueba de funcionamiento del brazo servo PWM</p>

Seteo de corriente inrush del driver A4988 **Medición de 400 mVdc para protección de 1 A del motor de paso de la banda transportadora**



Cableado y prueba de la banda transportadora con el driver potencia

Enlace de comunicación a la red WIFI del microcontrolador ESP32

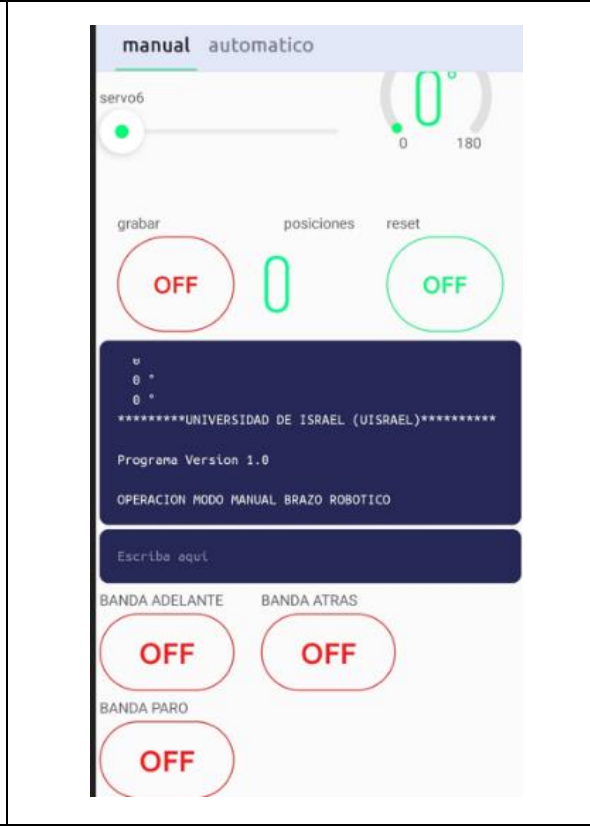
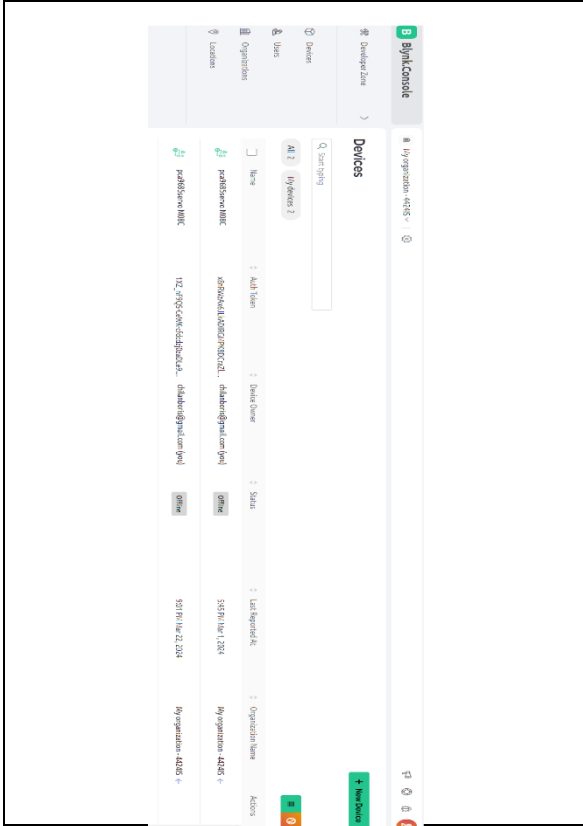


```

46
47 /* Fill in information from your Blynk Template here */
Output Serial Monitor x
Not connected. Select a board and a port to connect automatically.
09:24:30.115 -> A)*****5*****!!r*****r*****Hold the b
09:25:19.506 -> [50754] Connecting to WiFi: BCHILANMEGACOM_2GEXT
09:26:09.565 -> [100772] Connecting to WiFi: BCHILANMEGACOM_2GEXT
09:26:59.549 -> [150788] Connecting to WiFi: BCHILANMEGACOM_2GEXT
09:27:49.592 -> [200800] Connecting to WiFi: BCHILANMEGACOM_2GEXT
09:28:39.590 -> [250817] Connecting to WiFi: BCHILANMEGACOM_2GEXT
    
```

Web BLYNK con el template de programación

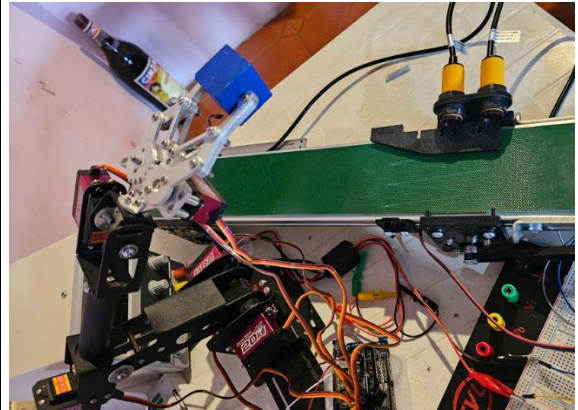
Visualización del menú modo manual del brazo robótico y banda transportadora con grabación de trayectoria.



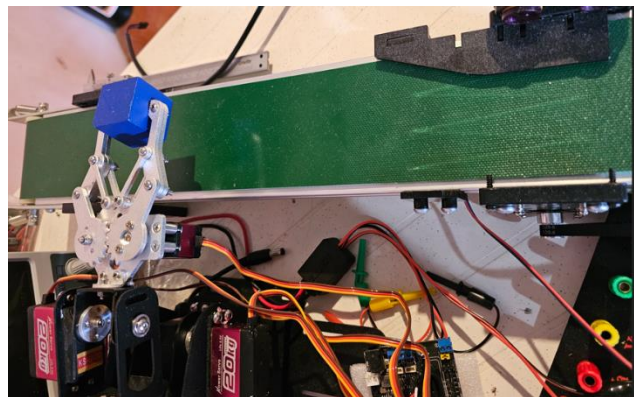
Visualización del menú modo automático del brazo robótico y banda transportadora con contador de objeto, velocidad y parada de emergencia



Ejecución del brazo robótico con la banda transportadora en modo de fábrica



Modo automático de proceso de simulación de maqueta de una fábrica 4.0



CONCLUSIONES

Se ha demostrado que la integración de la plataforma Blynk en el control del brazo robótico y la banda transportadora permite un control remoto eficiente y conveniente. Los usuarios pueden operar el sistema desde cualquier lugar con acceso a Internet, lo que aumenta la flexibilidad y la accesibilidad.

El sistema desarrollado tiene un gran potencial en entornos educativos para enseñar conceptos de robótica, automatización y control de procesos. La facilidad de uso de la interfaz de usuario y la capacidad de control remoto lo convierten en una herramienta valiosa para la enseñanza y el aprendizaje práctico.

La aplicación Blynk proporciona una interfaz de usuario intuitiva y fácil de usar, lo que facilita la operación del sistema incluso para usuarios no técnicos. Esto lo hace especialmente adecuado para aplicaciones educativas, donde se busca una curva de aprendizaje baja para los estudiantes.

El sistema ha demostrado una capacidad de control precisa y confiable tanto del brazo robótico como de la banda transportadora. Los movimientos son suaves y se ejecutan según las instrucciones proporcionadas a través de la aplicación Blynk, lo que garantiza un funcionamiento seguro y eficiente.

RECOMENDACIONES

Integrar la programación y la codificación en el currículo educativo asociado con el proyecto. Proporciona a los estudiantes oportunidades para escribir y modificar código para controlar el brazo robótico y la banda transportadora, lo que les ayudará a desarrollar habilidades.

Proporcionar a los estudiantes la oportunidad de explorar diferentes configuraciones y parámetros del sistema. Esto les permitirá comprender mejor los principios de la robótica y la automatización, así como experimentar con nuevas ideas e innovaciones.

Mantener el sistema actualizado con las últimas tecnologías y características para garantizar su relevancia y utilidad en el entorno educativo. Considerar la retroalimentación de los usuarios y realiza mejoras iterativas según sea necesario.

Realizar evaluaciones periódicas para medir el impacto educativo del sistema en los estudiantes. Esto puede incluir encuestas, pruebas de conocimiento y observaciones cualitativas para comprender cómo el proyecto está contribuyendo al aprendizaje y al desarrollo de habilidades de los estudiantes.

BIBLIOGRAFIA

- Verónica A. G.. (2011). sistema de control inalámbrico micro–robots brazo robótico, Masterado, UNIVERSIDAD TECNICA DE AMBATO, obtenido de <https://repositorio.uta.edu.ec/handle/123456789/78>
- Javier M. (2014). *Diseño del Sistema de control basado en Software libre para un brazo robótico de seis grados de libertad con funcionalidad de Mecanizado y paletizado*, Masterado, Universidad de las Fuerzas Armadas ESPE-L., obtenido de https://rraae.cedia.edu.ec/Record/ESPE_4ee1abe7cdf962894aa0569efbb5fa73
- Juan D.C.A. (2020). *Sistema integrado para la operación de un brazo robótico teleoperado en tiempo real mediante la plataforma Firebase con el uso de dispositivos móviles*, Masterado, Universidad Tecnológica de Israel, obtenido de <https://repositorio.uisrael.edu.ec/handle/47000/2395>
- Martínez M. (2017). *Sistema automatizado para transportar, clasificar y cuantificar piezas de tagua por color y forma*, Masterado, Universidad Tecnológica de Israel, obtenido de <https://repositorio.uisrael.edu.ec/handle/47000/1436>
- Aurelio C., ACADENAS (2020). *Como funciona circuito regulador pwm velocidad de motor*. <https://www.youtube.com/watch?v=A4lr6NNtD20>
- Michael K., Michael Klements (2020). *How To Correctly Set The Motor Current Limit On An A4988 Stepper Motor Driver*. <https://www.youtube.com/watch?v=OpaUwWouyE0>
- Edison V. editronikx (2022). *controla 16 SERVOS con la tarjeta PCA 9685 PWM DRIVER con ARDUINO o realiza PWM con 12 bits*. <https://www.youtube.com/watch?v=JUMyttiOGyE&t=463s>
- N.N. Programador Novato (2022). *Instalar ESP32 en Arduino IDE*. <https://www.youtube.com/watch?v=l-NW9qTfIUy>
- N.N. Puente Digital México (2021). *Curso Arduino e introducción a la robótica - Programación de brazo robótico*. <https://www.youtube.com/watch?v=qAoLhB1WF1M>

N.N. Sritu Hobby (2022). *How to assemble and control a 4 DOF robot mechanical arm kit with Arduino | Step by step.*
<https://www.youtube.com/watch?v=Q9JOKQaIR1w&t=20s>

N.N. SME DERHADUN (2023). *Control Multiple Servo Motors with New Blynk 2.0: A Step-by-Step Guide to 5 Servo Angle Control.*
<https://www.youtube.com/watch?v=DBrLP0DQazc>

N.N. Tech StudtCell (2021). *Home Automation using NodeMCU ESP8266 and Blynk 2.0 with real-time feedback | IoT Projects 2021.*
https://www.youtube.com/watch?v=CpUVssHPm_s&t=33s

Garcia. J. The profesor García (2023). *Multi - Tarea en Tiempo Real y en el CORE 2 Esto si es programación del ESP32.*
<https://www.youtube.com/watch?v=3yTW1HPUOpM&t=780s>

Jhimmy A. Jadsa Tech (2022). *nuevo blynk 2.0 - como usar blynk app blynk cloud - esp32 crear proyecto paso a paso.*
<https://www.youtube.com/watch?v=HRGQQATYJCQ&t=533s>

ANEXOS

ANEXO 1

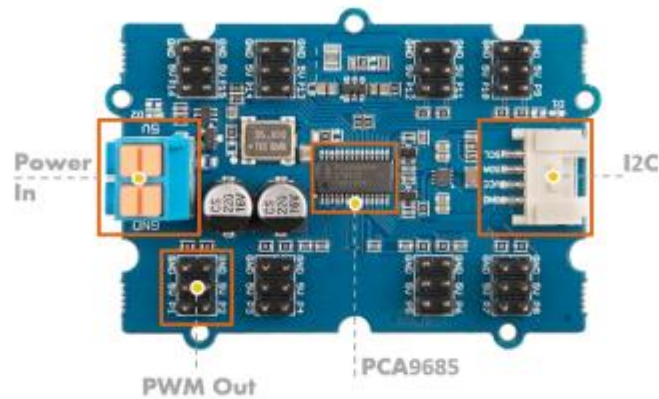
DRIVER DE POTENCIA SERVO

Grove - 16 Channel PWM Driver (PCA9685)



The Grove - 16-Channel PWM Driver is based on NXP PCA9685, which is a 16-Channel, 12 bit I2C PWM driver. This board can drive up to 16 servos with the external power supply. You can control this board with Arduino easily via the I2C Grove interface. In addition, you can use this board as a LED controller.

Pin Out



Warning

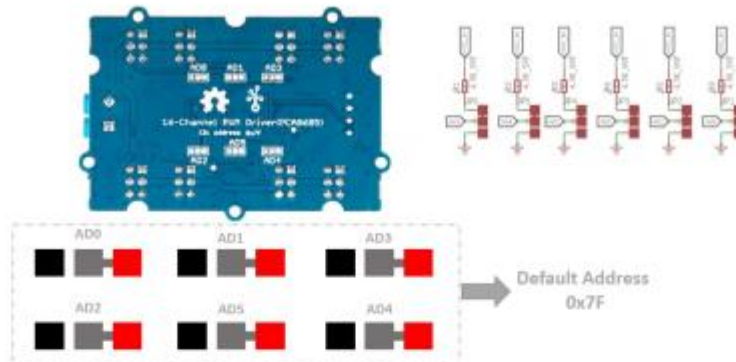
Hot swap is not supported, you may want to disconnect arduino from the power source before any replacement or change.

Hardware Detail

I2C Interface

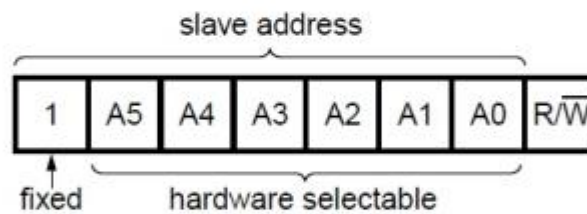
This board uses the I2C interface to allow the on-board MCU to communicate with the host computer.

GND: connect this module to the system GND
 VCC: you can use 5V or 3.3V for this module
 SDA: I2C serial data
 SCL: I2C serial clock



[https://files.seeedstudio.com/wiki/Grove-16-Channel_PWM_Driver-PCA9685/img/i2c_ad.jpg]

As shown in the figure above, all address pads are connected to the high level by default. The address rules are as shown below:



The address consists of 7 bits and the highest bit is fixed at 1. When we are converting the address to a hexadecimal number, we need to add a 0 to the left of the highest bit.

So the default address is **111 1111**, when we add a 0, it becomes **0111 1111** which is 0x7f.

And if we connect all the address pads to the ground, it turns to be **100 0000**, when we add a 0, it becomes **0100 0000**. Which is 0x40.

ANEXO 2

MANUAL DE CONFIGURACION DEL SENSOR INFRAROJO DE DISTANCIA

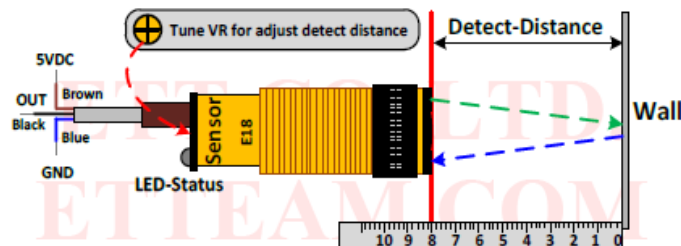
Manual of IR Sensor Switch E18-D80NK-N

IR-Sensor Switch E18 This is Sensor Infrared device for distance detection that can be adjusted in the range of 6 cm -80 cm.; and Output is Logic TTL; 0 (GND) and 1 (5V).

- Specifications**
- Adjust distance detection in the range of 6 cm -80 cm. by Adjustable VR and display the status by LED
 - Sensing device should be opaque material or any material that allows less light to pass through; black color is the best because Sensor device works well by using reflection of Infrared
 - OUTPUT is Open Collector; it has to connect R 10 K Pull Up at Out Putt
 - Signal Output is Digital TTL; 0 = GND and 1 = 5V
 - Use Power Supply DC 5V Current 100mA

How to setup distance detection: Before using, it has to setup preferable distance detection for using with Sensor as follows;

- 1) Provide 5V Power Supply (brown cable) and GND (blue cable) to Sensor
- 2) Turn the head of Sensor upright to the ground or wall (it is the best if ground or wall is black color)
- 3) Measure the preferable distance detection from ground or wall to the head of Sensor by ruler; and hold Sensor at the preferable position to detect for awhile
- 4) Adjust VR at the end of Sensor. Look at the change of LED at the end of Sensor as described below;



- If LED is OFF (OUTPUT = 1), please adjust VR in a clockwise direction until LED becomes ON (OUTPUT = 0) and then stop adjusting VR. The position that LED changes the state is the specified distance detection. This is conditional operation; if the distance of Sensor is less than or equal to the distance detection, LED Status is ON and OUTPUT becomes Logic 0; but if the distance of Sensor is greater than the distance detection, LED Status is OFF and OUTPUT becomes Logic 1 instead.

- If LED is ON (OUTPUT = 0), please adjust VR in an anticlockwise direction until LED becomes OFF (OUTPUT = 1) and then stop adjusting VR. The position that LED changes the state is the specified distance detection. This is conditional operation; if the distance of Sensor is greater than or equal to the distance detection, LED Status is OFF and OUTPUT becomes Logic 1; but if the distance of Sensor is less than the distance detection, LED Status is ON and OUTPUT becomes Logic 0 instead.

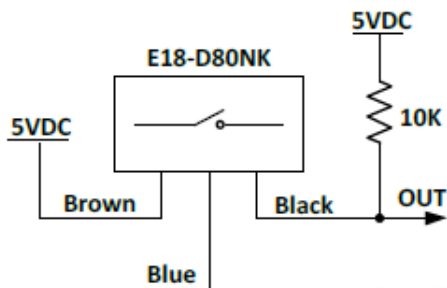
- 5) Test the operation of Sensor by moving Sensor. When the head of Sensor moves and passes the specified distance detection, LED of Sensor is lit up if the distance of Sensor is less or equal to the specified distance detection; but LED is OFF if the distance of Sensor is greater than or equal to the specified distance detection. If it does not accord with any conditional operation described above, it means that it fails to setup any distance detection for Sensor.

Referred to experiment in use, it found that color of ground or wall or any material that is used to reflect to Sensor is not enough dark. If the wall that is used to reflect is light color, the least distance detection of Sensor is also higher; so, the specified distance detection of user is lower than the least distance detection of Sensor. In this case, it should use wall with the dark color or it may setup the

distance detection higher, depend on material of user. User has to test and setup distance detection by self because each color of wall that reflects to Sensor is different; and finally, user needs to return to step 1-5. Referred to experiment, the least distance detection of the black wall that can reflect to Sensor is 6 cm; the operating result accords with step 5, it means that it succeeds and Sensor is ready to use and connect.

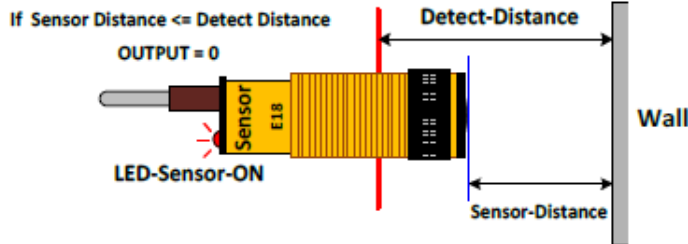
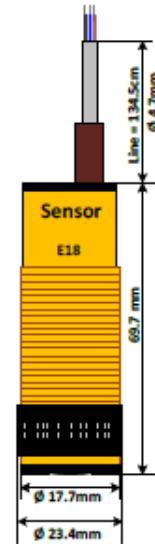
How to use Sensor after setup distance detection

Please look at the circuit below and connect Sensor with Connectors according to the specified color; Brown Cable is 5VDC Power Supply, Blue Cable is GND, and Black Cable is OUTPUT(TTL). Next, please look at the conditional operation of Sensor to write program correctly.

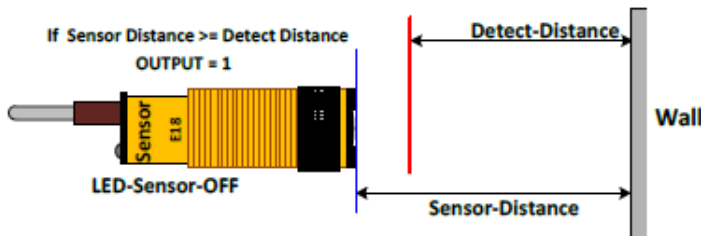


Connector and Circuit

Dimensions of Sensor



When distance of Sensor \leq the specified distance detection, LED Status is ON and OUTPUT = 0



When distance of Sensor \geq the specified distance detection, LED Status is OFF and OUTPUT = 1

ANEXO 3

DATOS TECNICOS DEL MOTOR DE PASO DE LA BANDA TRANSPORTADORA

2-Phase Hybrid Stepping Motor

1.8°

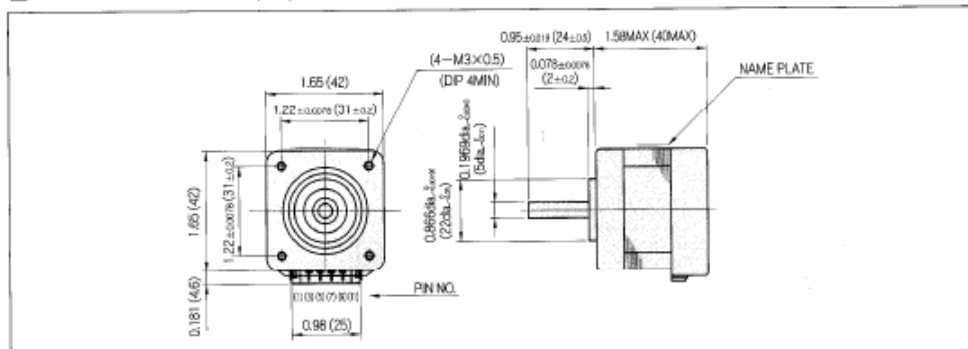
KH42 series 800 type

HIGH TORQUE, LOW VIBRATION AND LOW NOISE

STANDARD SPECIFICATIONS

MODEL	UNIT	KH42JM2				
		-801	-802	-803	-851	-852
DRIVE METHOD	———	UNI-POLAR			BI-POLAR	
NUMBER OF PHASES	———	2			2	
STEP ANGLE	deg./step	1.8			1.8	
VOLTAGE	V	3.96	5.93	9.6	5.61	8.25
CURRENT	A/PHASE	1.2	0.79	0.48	0.85	0.55
WINDING RESISTANCE	Ω/PHASE	3.3	7.5	20	6.6	15
INDUCTANCE	mH/PHASE	2.4	5.1	15	9.7	20.5
HOLDING TORQUE	kgf · cm	2.4	2.4	2.4	3.2	3.2
	oz · in	33	33	33	44	44
DETENT TORQUE	gf · cm	150	150	150	150	150
	oz · in	2.1	2.1	2.1	2.1	2.1
ROTOR INERTIA	g · cm ²	48	48	48	48	48
	oz · in ²	0.26	0.26	0.26	0.26	0.26
WEIGHTS	g	260	260	260	260	260
	lb	0.57	0.57	0.57	0.57	0.57
INSULATION CLASS	———	JIS Class E (120°C 248° F) (UL VALUE : CLASS B-130°C 266° F)				
INSULATION RESISTANCE	———	500VDC 100MΩmin.				
DIELECTRIC STRENGTH	———	500VAC 50HZ 1min.				
OPERATING TEMP. RANGE	°C	0 to 50				
ALLOWABLE TEMP. RISE	deg.	70				

DIMENSIONS unit = inch (mm)

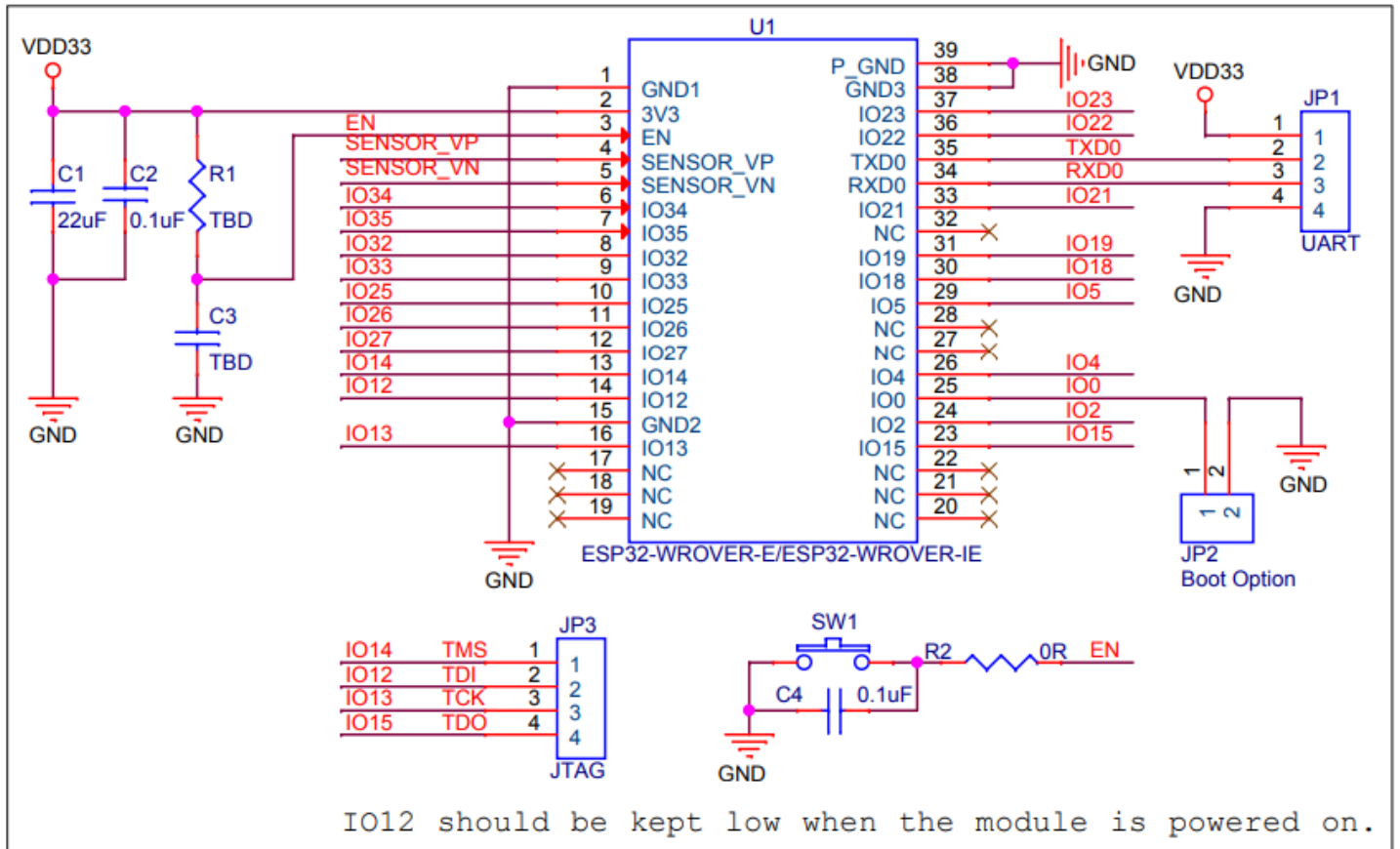


ANEXO 4

ESQUEMATICO ELECTRICO DEL ESP32 WROVER

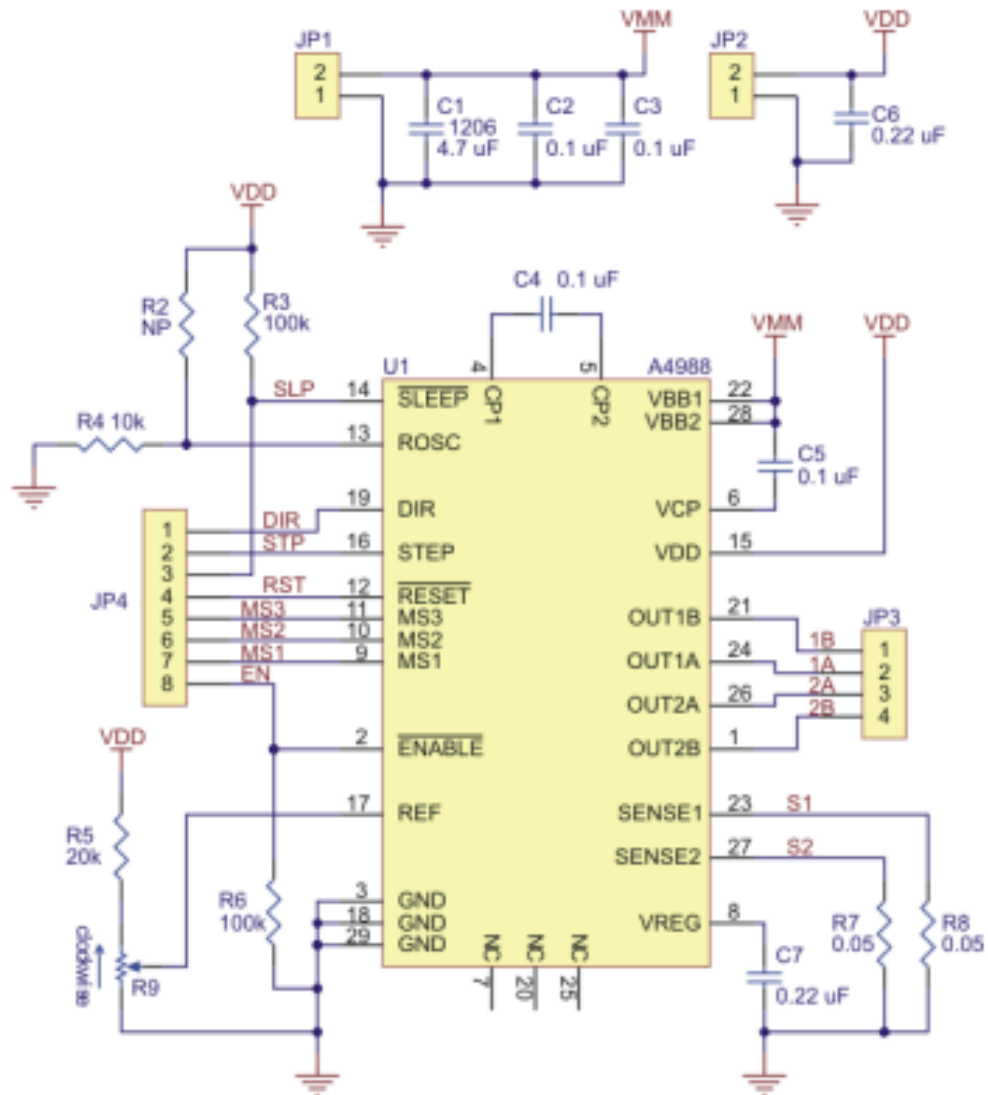
8 Peripheral Schematics

This is the typical application circuit of the module connected with peripheral components (for example, power supply, antenna, reset button, JTAG interface, and UART interface).



ANEXO 5

ESQUEMATICO ELECTRICO DEL DRIVER DE POTENCIA PARA MOTOR DE PASO



Schematic diagram of the md09b A4988 stepper motor driver carrier.

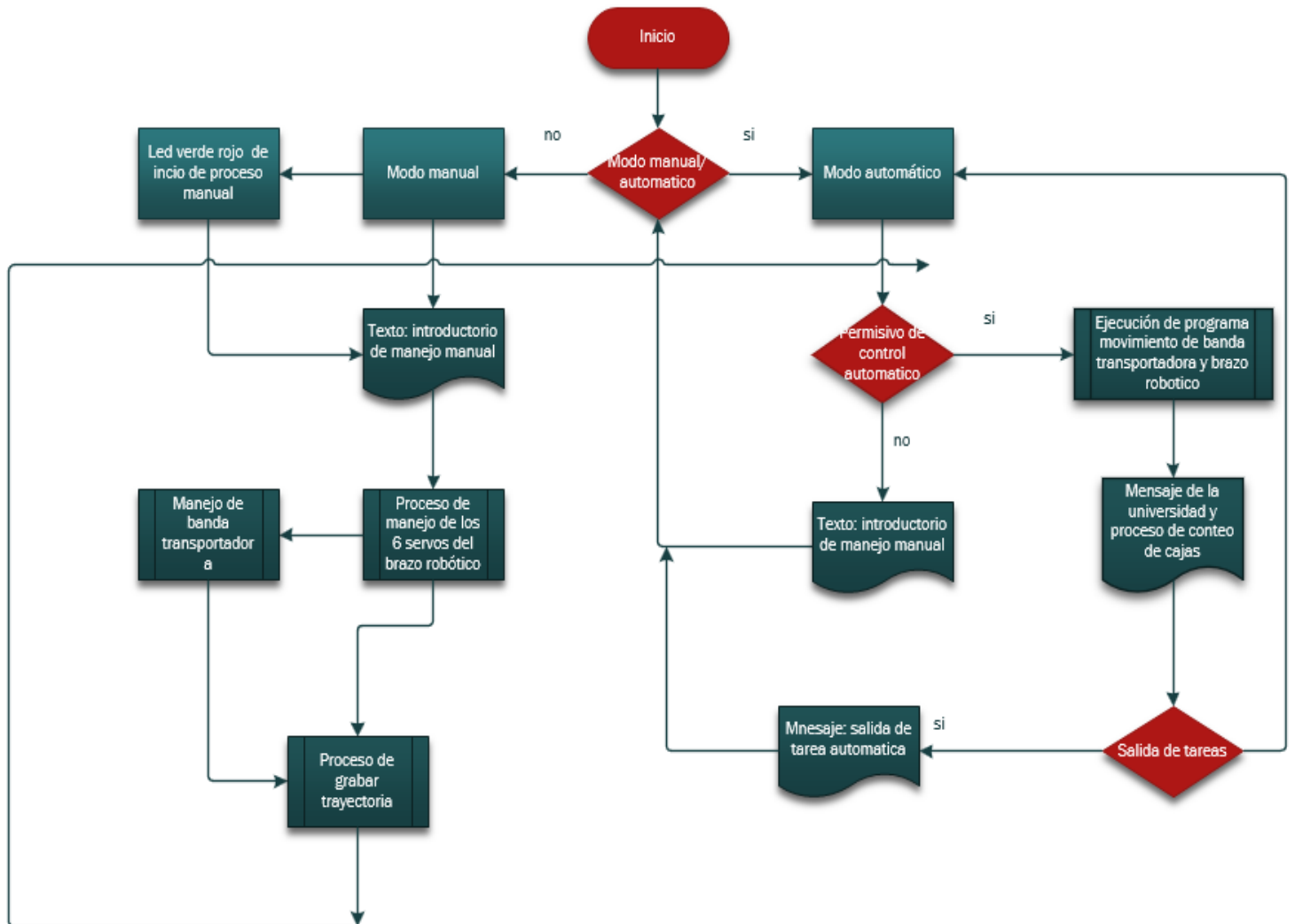


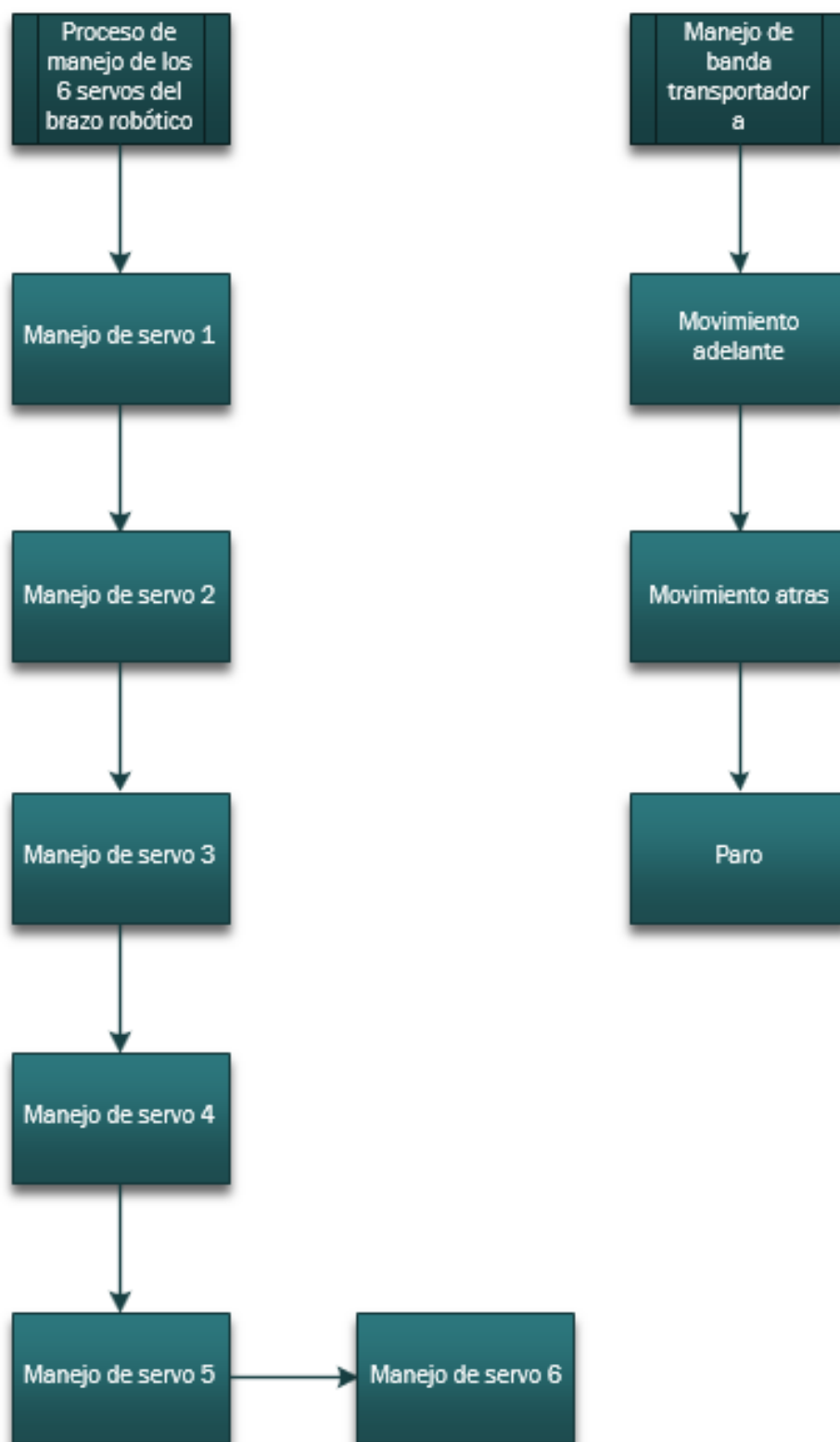
Note: This board is a drop-in replacement for our original A4988 stepper motor driver carrier.

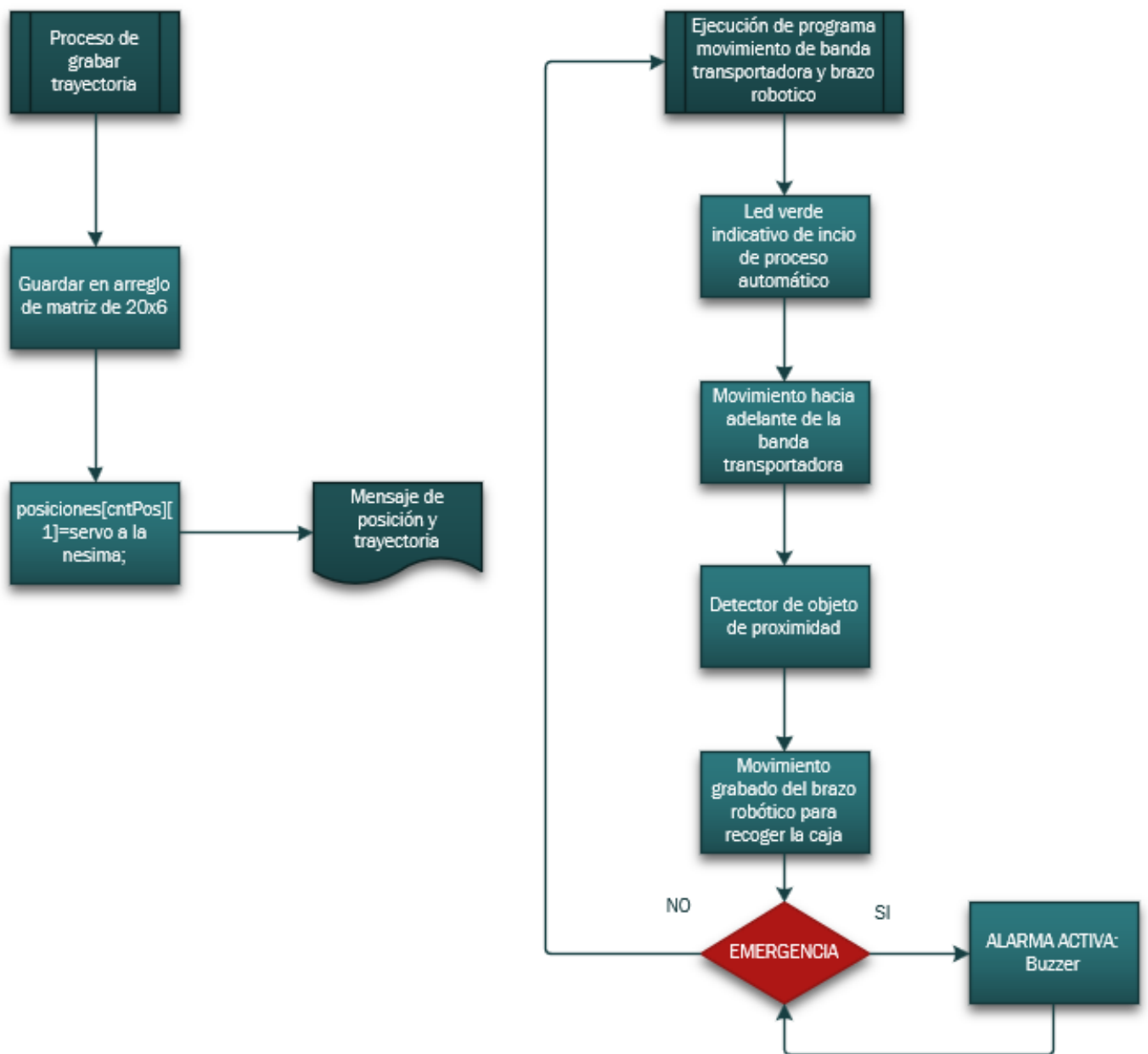
[Documentation on producer website.](#)

ANEXO 6

DIAGRAMA DE FUJO BAJO LA PLATAFORMA VISIO OFFICE







ANEXO 7

CODIGO DE PROGRAMACIÓN ARDUINO IDE v2.0

```
/*
*****
Blynk is a platform with iOS and Android apps to control
ESP32, Arduino, Raspberry Pi and the likes over the Internet.
You can easily build mobile and web interfaces for any
projects by simply dragging and dropping widgets.

Downloads, docs, tutorials: https://www.blynk.io
Sketch generator:          https://examples.blynk.cc
Blynk community:         https://community.blynk.cc
Follow us:                https://www.fb.com/blynkapp
                          https://twitter.com/blynk_app

Blynk library is licensed under MIT license
*****
Blynk.Edgent implements:
- Blynk.Inject - Dynamic WiFi credentials provisioning
- Blynk.Air    - Over The Air firmware updates
- Device state indication using a physical LED
- Credentials reset using a physical Button
*****/

#include <Stepper.h>
#include "Adafruit_PWMServoDriver.h"
#include <HardwareSerial.h>
#include <HardwareSerial.h>

//HardwareSerial SerialPort(0); // use UART1

#include "esp_task_wdt.h"
Adafruit_PWMServoDriver pwmT0 = Adafruit_PWMServoDriver(0x60);
#define MIN_PULSE_WIDTH 600
#define MAX_PULSE_WIDTH 2600
#define FREQUENCY 50

//#define RXD0 3
//#define TXD0 1
const byte ledpin = 13;
const byte ledpin2=12;
const byte buzzer=14;
//variables de la banda transportadora
const int dirPin =33;
const int stepPin = 32;
const int detecBox=35;
```

```

const int stepsPerRevolution = 400;

/* Fill in information from your Blynk Template here */
/* Read more: https://bit.ly/BlynkInject */
#define BLYNK_TEMPLATE_ID          "TMPxxxxxx"
#define BLYNK_TEMPLATE_NAME        "Device"
#define BLYNK_TEMPLATE_ID "TMPL2Cp-0q65q"
#define BLYNK_TEMPLATE_NAME "pca9685servo"
#define BLYNK_AUTH_TOKEN "ssr303mfdCRXLQZwet2cf_B9DmbTLuOv"

#define BLYNK_FIRMWARE_VERSION     "0.1.0"

#define BLYNK_PRINT Serial
//#define BLYNK_DEBUG

#define APP_DEBUG

// Uncomment your board, or configure a custom board in Settings.h
//#define USE_ESP32_DEV_MODULE
//#define USE_ESP32C3_DEV_MODULE
//#define USE_ESP32S2_DEV_KIT
//#define USE_WROVER_BOARD
//#define USE_TTGO_T7
//#define USE_TTGO_T_OI

#include "BlynkEdgent.h"
#include <BlynkSimpleEthernet.h>
int manual;
int automatico;
int servo1,servo2,servo3,servo4,servo5,servo6;
int anchopulso;
int
analog_value,analog_value2,analog_value3,analog_value4,analog_value5,anal
og_value6;
int led;
int led1;
int grabarPos,cntPos,resetPos;
int posiciones[20][20];
int pos_analog[20][20];
int anaVal[20];
int grado1,grado2,grado3,grado4,grado5,grado6;
int flg_grabado,flg_fabrica;
int p,salir_auto,trayectoria;
int16_t cnt,cnt_servos;
int delayServos;
int show,buttonStop,buttonADELANTE,buttonATRAS,buttonPARO;

```

```

int flgCaja,tempAngulo,tmpAnalogValue,cntCaja;
//char boxchar=' ';
int opcBanda=0;
//int cntCaja=0;

//variable banda transportadora////////////////////////////////////
//const int e18_sensor = 7;
//const int stepsPerRevolution = 200; // cha
//Stepper myStepper(stepsPerRevolution, 15, 2, 33, 32);
////////////////////////////////////
//WidgetTerminal terminal(V100);
//HardwareSerial SerialPort(0); // use UART2

BLYNK_WRITE(V10){
  resetPos=param.asInt();
  cntPos=0;
  Blynk.virtualWrite(V9,cntPos);
  //encerar la matriz de posiciones
  for (int i=1; i<20; i++)
    for(int j=1;j<20;j++){
      posiciones[i][j]=0;
      //Blynk.virtualWrite(V100, posiciones[i][j]);
    }
  for (int i=1; i<20; i++)
    anaVal[i]=0;

  Blynk.virtualWrite(V100, "clr");
  Blynk.virtualWrite(V100, "*****UNIVERSIDAD DE ISRAEL
(UISRAEL)*****\n");
  Blynk.virtualWrite(V100, "Programa Version 1.0\n");
  Blynk.virtualWrite(V100, "OPERACION MODO MANUAL BRAZO ROBOTICO");
  Blynk.virtualWrite(V100, "\nPOSICION:", cntPos, "\nGrados:");
  for(int j=1;j<=6;j++)
    Blynk.virtualWrite(V100, " ", posiciones[cntPos][j], "°");

}

BLYNK_WRITE(V8){

  grabarPos=param.asInt();

  if (grabarPos==1){
    while(grabarPos==0){};
    flg_grabado=1;
    cntPos= cntPos+1;
    Blynk.virtualWrite(V9,cntPos);
    posiciones[cntPos][1]=servo1;

```

```

        posiciones[cntPos][2]=servo2;
        posiciones[cntPos][3]=servo3;
        posiciones[cntPos][4]=servo4;
        posiciones[cntPos][5]=servo5;
        posiciones[cntPos][6]=servo6;
        Blynk.virtualWrite(V100,"\nPOSICION:",cntPos,"\nGrados:");
        for(int j=1;j<=6;j++)
            Blynk.virtualWrite(V100," ",posiciones[cntPos][j],"°");
    }
}

```

```

BLYNK_WRITE(V0){// servo1 base robot
    servo1 = param.asInt();
}
BLYNK_WRITE(V3){// servo2
    servo2 = param.asInt();
}
BLYNK_WRITE(V4){// servo4
    servo3 = param.asInt();
}
BLYNK_WRITE(V5){// servo5
    servo4 = param.asInt();
}
BLYNK_WRITE(V6){// servo6
    servo5 = param.asInt();
}
BLYNK_WRITE(V7){// servo6
    servo6 = param.asInt();
}

```

```

BLYNK_WRITE(V1){

    led = param.asInt();
    //digitalWrite(ledpin,led);
    Blynk.virtualWrite(V1,led);

    manual=led;
    digitalWrite(ledpin,manual);
    Blynk.virtualWrite(V100, "clr");
    Blynk.virtualWrite(V100,"*****UNIVERSIDAD DE ISRAEL
(UISRAEL)*****\n");
    Blynk.virtualWrite(V100,"Programa Version 1.0\n");
}

```

```

    Blynk.virtualWrite(V100,"OPERACION MODO MANUAL BRAZO ROBOTICO");
}

BLYNK_WRITE(V2){
    int led1 = param.asInt();

    Blynk.virtualWrite(V2,led1);
    automatico=led1;
    digitalWrite(ledpin2,automatico);
}
BLYNK_WRITE(V11){
    int var = param.asInt();
    salir_auto= var;
    flg_fabrica=automatico=0;
    show=1;
    Blynk.virtualWrite(V101, "clr");
    Blynk.virtualWrite(V101,"SALIENDO MODO AUTOMATICO...");
    delay(200);
    digitalWrite(ledpin,0);
    digitalWrite(ledpin2,0);
}
BLYNK_WRITE(V12){
    int var = param.asInt();

    delayServos=var;

}

BLYNK_WRITE(V13){
    int var3 = param.asInt();

    buttonStop=var3;
    Blynk.virtualWrite(V101, "clr");
    Blynk.virtualWrite(V101,"PELIGRO!!!. ALARMA!!!...\n");
    delay(200);

}

//////////control de banda
blynk//////////
    BLYNK_WRITE(V14){
    int var4 = param.asInt();

    opcBanda=1;
    //Serial.println("a");
    Blynk.virtualWrite(V101, "clr");
    Blynk.virtualWrite(V101,"MOVIMIENTO BANDA ADELANTE\n");
    delay(200);
}

```

```

}
BLYNK_WRITE(V15){
int var5 = param.asInt();

    opcBanda=2;
    //Serial.println("b");
    Blynk.virtualWrite(V101, "clr");
    Blynk.virtualWrite(V101,"MOVIMIENTO BANDA ATRAS\n");
    delay(200);

}
BLYNK_WRITE(V16){
int var6 = param.asInt();

    opcBanda=0;
    //Serial.println("c");
    Blynk.virtualWrite(V101, "clr");
    Blynk.virtualWrite(V101,"BANDA DETENIDA\n");
    delay(200);

}
//////////////////////////////////////////////////FIN control de banda
blynk////////////////////////////////////

void setServo(uint8_t n_servo, int angulo) {
    int duty;
    duty=map(angulo,0,180,MIN_PULSE_WIDTH, MAX_PULSE_WIDTH);
    int valor_analogo = int(float(duty) / 1000000 * FREQUENCY * 4096);
    pwmT0.setPWM(n_servo, 0, valor_analogo);
}

TaskHandle_t Tarea0;
////////////////////////////////////SETUP DEL
PROGRAMA////////////////////////////////////
void setup()
{

    Wire.begin(21,22);
    Serial.begin(115200);
    //Serial.begin(9600, SERIAL_8N1, RXD0, TXD0);
    xTaskCreatePinnedToCore(loop0, "Tarea_0", 3000, NULL, 1, &Tarea0, 0);
    // vTaskDelay(2);
    pinMode(ledpin, OUTPUT);
    pinMode(ledpin2, OUTPUT);
    pinMode(buzzer, OUTPUT);
    //seteos de la banda transportadora
    pinMode(stepPin, OUTPUT);
    pinMode(dirPin, OUTPUT);
}

```

```

pinMode(detecBox, INPUT);
////////////////////////////////////
pwmT0.begin();
pwmT0.setPWMFreq(FREQUENCY);
cntPos>manual=automatico=0;
Blynk.virtualWrite(V9,cntPos);
for (int i=1; i<20; i++)
    for(int j=1;j<20;j++)
        posiciones[i][j]=0;
for (int i=1; i<20; i++)
    anaVal[i]=0;

delay(100);
cntCaja=opcBanda=flgCaja=cntCaja=buttonStop=salir_auto=flg_fabrica=flg_
grabado=0;
delayServos=5;
show=1;

//myStepper.setSpeed(120);
//initialize the serial port:
//Serial.begin(9600);
//pinMode (e18_sensor, INPUT);

BlynkEdgent.begin();
// disableCore0WDT();
//esp_task_wdt_delete(NULL);
}

void loop() {
////////////////////////////////+++++++MODO MANUAL*****
while>manual==1){//tarea modo manual

digitalWrite(ledpin>manual);
digitalWrite(ledpin2,0);
digitalWrite(buzzer,0); //alarma en brazo

setServo(0,servo1);
setServo(1,servo2);
setServo(2,servo3);
setServo(3,servo4);
setServo(4,servo5);
setServo(5,servo6);

if(automatico==1)
   >manual=0;

```



```

        //vTaskDelay(2);
        BlynkEdgent.run();
    }////////////////////////////////////cierre de tarea
modo manual////////////////////////////////////

////////////////////////////////////+++++++MO
DO AUTOMATICO*****
    while(automatico==1){
        //vTaskDelay(2);

        if(flg_grabado==1){
            Blynk.virtualWrite(V101, "clr");
            Blynk.virtualWrite(V101,"*****UNIVERSIDAD DE ISRAEL
(UIRAEL)*****\n");
            Blynk.virtualWrite(V101,"Programa Version 1.0\n");
            Blynk.virtualWrite(V101,"OPERACION MODO AUTOMATICO FABRICA
4.0");
            delay(3000);
            digitalWrite(ledpin,0);
            digitalWrite(ledpin2,automatico);
            delay(1000);
            Blynk.virtualWrite(V101, "clr");
            Blynk.virtualWrite(V101,"\nPROBANDO OPERACION ROBOT EN ESTADO
REPOSO.....");
            flg_grabado=0;
            flg_fabrica=1;
            opcBanda=1; //transportador de banda en funcionamimoto
        }
        else{
            Blynk.virtualWrite(V101, "clr");
            Blynk.virtualWrite(V101,"PRIMERO DEBE GRABAR LAS POSICIONES DEL
BRAZO!!!!\n");
            delay(1000);
            digitalWrite(ledpin,0);
            digitalWrite(ledpin2,0);
            automatico=0;
        }
        while(flg_fabrica==1){
            //opcBanda=1; //transportador de banda en funcionamimoto
            if(digitalRead(detecBox)==0){
                opcBanda=0;
                flgCaja=1;
                cntCaja=cntCaja+1;
            }
            if(flgCaja==1){
                //opcBanda=0;
                //flgCaja=0;

```

```

        Blynk.virtualWrite(V101,"ejecutando\n");
        Blynk.virtualWrite(V101,"CAJA:",cntCaja,"\n");
        for(trayectoria=1;trayectoria<=cntPos;trayectoria++)
            for( cnt_servos=1;
cnt_servos<=6;cnt_servos++){//contadores de servos del robot
                if(posiciones[trayectoria][cnt_servos]<posiciones[
trayectoria+1][cnt_servos]){//trayectoria positiva
                    for(tempAngulo=posiciones[trayectoria][cnt_se
rvos];
tempAngulo<=posiciones[trayectoria+1][cnt_servos];tempAngulo++){ //recor
rido del angulo
                        anchopulso = map(tempAngulo, 0, 180,
MIN_PULSE_WIDTH, MAX_PULSE_WIDTH);
                        tmpAnalogValue = int(float(anchopulso) /
1000000 * FREQUENCY * 4096);
                        //base[cnt_servos-
1]=analogue_value;
                        pwmT0.setPWM(cnt_servos-
1,0,tmpAnalogValue);
                        //Blynk.virtualWrite(V101,analog_value,"\n");
                        delay(delayServos);
                        while(buttonStop==1){BlynkEdgent.run();};//b
ucle de emergencia
                        BlynkEdgent.run();
                    }//cierre del recorrido del angulo
                } //cierre de logica de trayectoria positiva
            else
if(posiciones[trayectoria][cnt_servos]>posiciones[trayectoria+1][cnt_serv
os]){//trayectoria negativa
                for(
tempAngulo=posiciones[trayectoria][cnt_servos];
tempAngulo>=posiciones[trayectoria+1][cnt_servos];tempAngulo--
){ //recorrido del angulo
                    anchopulso = map(tempAngulo, 0, 180,
MIN_PULSE_WIDTH, MAX_PULSE_WIDTH);
                    tmpAnalogValue = int(float(anchopulso) /
1000000 * FREQUENCY * 4096);
                    //base[cnt_servos-1]=analogue_value;
                    pwmT0.setPWM(cnt_servos-1,0,tmpAnalogValue);
                    delay(delayServos);
                    while(buttonStop==1){
                        /* if(buttonStop==1){
                            tone(buzzer, 1000); // Send 1KHz
sound signal...
                            delay(1000);
                        }
                    }
                }
            }

```

```

else
    noTone(buzzer); // Send 1KHz

sound signal... */

    BlynkEdgent.run();
} //bucle de emergencia
// noTone(buzzer); // Send 1KHz sound

signal...

    BlynkEdgent.run();
} //cierre del recorrido del angulo
} //cierre de trayectoria negativa
else{//trayectorias iguales
    tempAngulo=posiciones[trayectoria][cnt_servos];
    anchopulso = map(tempAngulo, 0, 180,
MIN_PULSE_WIDTH, MAX_PULSE_WIDTH);
    tmpAnalogValue = int(float(anchopulso) /
1000000 * FREQUENCY * 4096);
    //base[cnt_servos-1]=analogue_value;
    pwmT0.setPWM(cnt_servos-
1,0,tmpAnalogValue);

} //cierre de trayectorias iguales

} //cierre de contadores del servo del motor

while(digitalRead(detecBox)==0){}

//vTaskDelay(2);
} //caja encontrada

    opcBanda=1; //transportador de banda en funcionamieto
    flgCaja=0;
    BlynkEdgent.run();

} //cerrado ciclo de fabrica

if(manual==1)
    automatico=0;

```

```

    BlynkEdgent.run();

} //cerrado de modo automatico

//vTaskDelay(2);

    BlynkEdgent.run();
}
////////////////////////////////////*****TAREA EN CORE 1 banda
transportadora*****
void loop0(void *parameter){

    while(1==1){

switch (opcBanda) {
case 1://modo adelante
    digitalWrite(dirPin, LOW);
    digitalWrite(stepPin, HIGH);
    delayMicroseconds(2000);
    digitalWrite(stepPin, LOW);
    delayMicroseconds(2000);
    break;
case 2://modo atras
    digitalWrite(dirPin, HIGH);
    digitalWrite(stepPin, HIGH);
    delayMicroseconds(2000);
    digitalWrite(stepPin, LOW);
    delayMicroseconds(2000);
    break;
default:// para la banda
    digitalWrite(stepPin, LOW);
    break;
}

        while(buttonStop==1){
            digitalWrite(stepPin, LOW);
            tone(buzzer, 1000); // Send 1KHz sound signal...
            delay(1000);
            noTone(buzzer); // Send 1KHz sound signal...
            delay(1000);

```

```
    }  
    //BlynkEdgent.run();  
    vTaskDelay(10);  
  
    }  
    vTaskDelay(10);  
    BlynkEdgent.run();  
    //disableCore0WDT();  
}
```