

Maximiliano Passo Koziura

Ezequiel Nahuel Fernandez Zalazar

Grupo 7 - Araña

Especialidad en Computación, Escuela Técnica N°32 D.E. 14

4to 1ra: Proyecto Informático

Gonzalo Consorti

24 de Noviembre de 2023

funcionalidades

Al comienzo del desarrollo yo y Ezequiel decidimos pautar las funcionalidades que el proyecto debía de tener, primero que nada reservamos el proyecto con Consorti explicandole lo que planeábamos hacer y qué funcionalidades pensábamos tener, he de decir que fueron funcionalidades demasiado ambiciosas para el poco tiempo del que tenemos, probablemente en dos meses solo logremos poder hacer que camine con suerte, sin mencionar que los diseños que estuvimos viendo no estarían hechos para tener otra plataforma encima, por lo que se podría decir que desde un principio descartamos tal característica ya que no la tuvimos en cuenta para el desarrollo.

Sigo queriendo hacer un Curiosity >:(, sin embargo estas fueron las características básicas que decidimos que el proyecto debería tener

- Movimiento Adelante, Atrás y Costados:
 - Cada pata de la araña debe tener al menos dos servos para permitir movimientos multidireccionales.
 - Para moverse hacia adelante, las patas realizarán una secuencia de movimientos coordinados para avanzar.
 - Los movimientos hacia atrás se pueden lograr invirtiendo la secuencia de movimientos hacia adelante.
 - Los movimientos laterales pueden ser alcanzados mediante la coordinación adecuada de los servos en cada pata
- Movimiento Independiente de las Patas y la Parte Superior:
 - Cada pata debe ser controlada independientemente para permitir un movimiento coordinado.
 - La parte superior de la araña, que puede incluir el cuerpo y posiblemente brazos adicionales, debe moverse de manera independiente de las patas.

- Esto implica programar los servos de las patas y la parte superior por separado.
- Control a través de una Aplicación en App Inventor:
 - Diseña una interfaz de usuario en App Inventor que incluya botones o controles deslizantes para cada dirección de movimiento.
 - Utiliza un módulo Bluetooth en Arduino para establecer una conexión entre el Arduino y la aplicación en el teléfono.
 - Asociar cada botón en la aplicación con funciones en el código de Arduino que controlen los movimientos de las patas.
- Sensor de Proximidad para Evitar Choques:
 - Agrega un sensor de proximidad, como un sensor ultrasónico, al frente de la araña.
 - Programa el Arduino para leer constantemente los datos del sensor de proximidad.
 - Si el sensor detecta un obstáculo a una distancia predeterminada, detiene el movimiento o ajusta la dirección para evitar el choque.
- Optimización y Refinamiento:
 - Ajusta los ángulos y la velocidad de los servos para lograr movimientos suaves y realistas.
 - Experimenta con diferentes secuencias de movimientos para mejorar la eficiencia y la estabilidad del robot.
 - Implementa un control de velocidad para ajustar la velocidad de desplazamiento.
- Seguridad y Supervisión:
 - Implementa un mecanismo de parada de emergencia en caso de mal funcionamiento o condiciones peligrosas.

- Monitorea la temperatura de los servos para evitar el sobrecalentamiento

Posibles diseños

<https://www.thingiverse.com/search?q=spider+robot&page=4&type=things&sort=relevant>

<https://www.thingiverse.com/thing:5319247>

<https://www.thingiverse.com/thing:3379770>

<https://www.thingiverse.com/thing:3130196> → elegido

En sí todos los diseños son similares, son robots de 2 servos por pata, Al tener dos servos por pata, se logra un menor rango de movimiento y flexibilidad en comparación con diseños que utilizan 3 servos. Esto permite simular de manera más menos efectiva los movimientos naturales de una araña, sin embargo esto tiene una razón de ser, ya que no teníamos suficientes servos para que pudiera usar yo solo en mi proyecto y por más poseía la plata para comprar los servos, considere que no valía la pena gastar mas de 10 mil pesos en comprar 4 servos o más para un simple proyecto escolar.

Realmente los diseños no tienen mucho sobre lo que hablar, más teniendo en cuenta que al final en si no se respetó el diseño de ninguno de los ejemplos y se hizo prácticamente cualquier cosa prácticamente.

Cosas que tuvimos (mentira, salió todo mal, tendríamos que haberlo tenido en cuenta) en cuenta para este proyecto

Diseño Mecánico:

Estructura de las patas: Decide la cantidad de patas que tendrá tu araña y el diseño de cada pata. Considera la disposición de los servos y cómo interactúan para lograr el movimiento deseado.

Articulaciones y Conexiones: Planifica las articulaciones y conexiones entre los servos. Define cómo se conectarán los servos para lograr un movimiento coordinado.

Elección de Componentes:

Servomotores: Selecciona servos que sean adecuados para las dimensiones y el peso del robot. Asegúrate de que los servos sean lo suficientemente potentes para mover las patas de manera eficiente.

Arduino: Elige una placa Arduino compatible para controlar los servos y recibir comandos desde la aplicación.

Módulo Bluetooth: Si planeas controlar el robot mediante una aplicación, necesitarás un módulo Bluetooth para la comunicación inalámbrica entre el Arduino y el dispositivo móvil.

Diseño Electrónico:

Esquema Eléctrico: Crea un esquema eléctrico que muestre cómo se conectarán los servos, el Arduino y otros componentes eléctricos.

Fuente de Alimentación: Determina la fuente de alimentación para los servos y el Arduino. Asegúrate de proporcionar la potencia necesaria para un funcionamiento eficiente.

Programación:

Código Arduino: Desarrolla el código necesario para controlar los servos. Implementa funciones para los diferentes movimientos (adelante, atrás, lateral) y asegúrate de que los movimientos de las patas sean independientes del cuerpo.

Comunicación Bluetooth: Programa la comunicación Bluetooth para recibir comandos desde la aplicación.

App Inventor:

Diseño de la Aplicación: Diseña la interfaz de usuario de la aplicación en App Inventor. Crea botones o controles que envíen comandos de movimiento al Arduino a través del módulo Bluetooth.

Programación de la Aplicación: Desarrolla el código en App Inventor para enviar comandos de movimiento al robot.

Prototipado y Pruebas:

Impresión 3D o Fabricación de Piezas: Si decides utilizar impresión 3D, diseña y fabrica las piezas necesarias para la estructura del robot.

Montaje: Ensambla todas las piezas, conecta los servos y verifica la movilidad de las patas.

Pruebas Iniciales: Realiza pruebas iniciales para verificar que los movimientos son fluidos y que el control desde la aplicación funciona correctamente.

Sensor de Proximidad:

Selección del Sensor: Elige un sensor de proximidad adecuado para detectar obstáculos. Puedes optar por sensores ultrasónicos u otros tipos según tus necesidades.

Integración: Implementa la integración del sensor de proximidad en el código para evitar colisiones.

Ajustes y Optimización:

Ajuste de Ángulos: Experimenta con los ángulos de los servos para lograr movimientos suaves y naturales.

Optimización del Código: Realiza ajustes en el código para mejorar la eficiencia y respuesta del robot.

Documentación y Seguimiento:

Carpeta de Campo: Documenta cada paso del proyecto en una "Carpeta de Campo". Registra los problemas encontrados, soluciones implementadas y resultados de pruebas.

Tiempo y Recursos Disponibles:

Planificación del Tiempo: Estima el tiempo necesario para cada fase del proyecto y planifica adecuadamente.

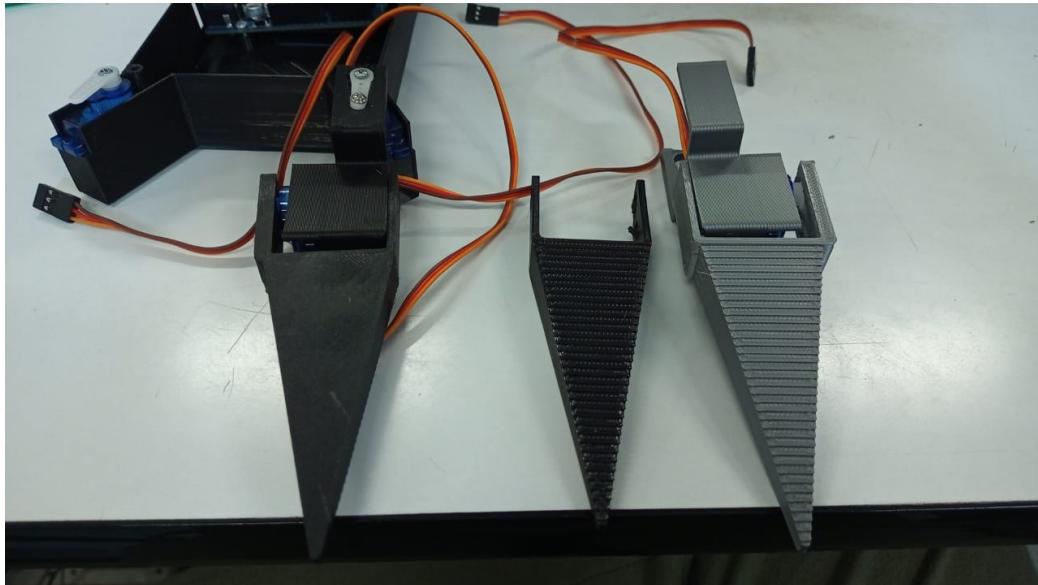
Recursos: Asegúrate de tener acceso a los recursos necesarios, como materiales de construcción, herramientas y equipos de prueba.

Comienzo del desarrollo

Comencé el proyecto haciendo un circuito en tinkerCAD para tener una referencia de la posición de los servos e ir viendo el movimiento de las patas.

Pero eso no fue todo, en realidad el primer problema que tuve fue hacer una buena coordinación con el mono, no fue difícil explicarle que necesitaba en sí pero cambió demasiado la idea original del diseño, bueno, supongamos que por más que tenga diferente el diseño es

funcional, bueno, no, como se va a mencionar a profundidad más adelante, tuve demasiados problemas con casi todo, en total ya fueron impresas 4 diseños de patas diferentes



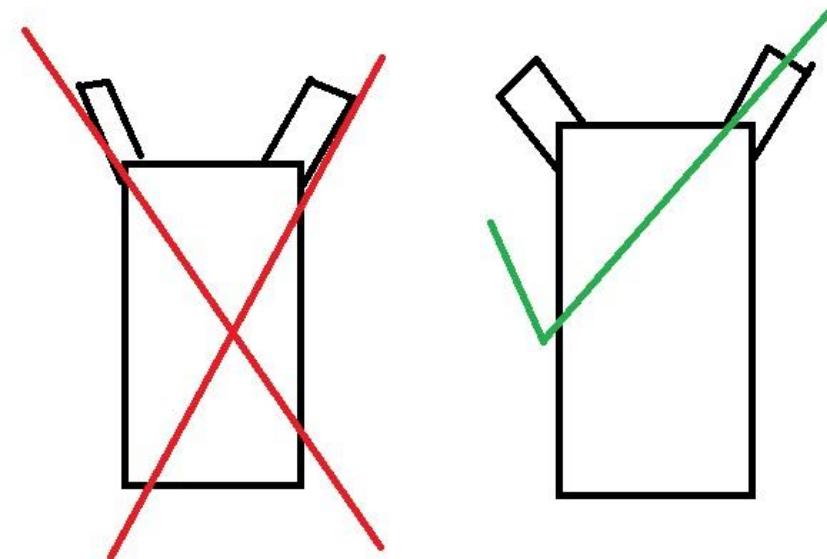
(falta una la cual fue desechada)

y hasta la fecha (17/11) solo tengo una pata funcional, que en si no es tan funcional, si no que para que agarre bien tuve que atornillar el servo a las piezas, pero en fin.

Volviendo al inicio, tuve que concordar por bastante tiempo con el Mono, a pesar de haber disponido de sus servicios bastante tempranamente tuve demasiados retrasos, y probablemente estaria muchisimo mas atrasado si no hubiese dedicado varios días libres de escuela que tenia para ir con el mono, tuve que ir multiples veces en horario extracurricular y muchas otras en horario de escuela.

La primera pieza que diseñó el mono fue la base en la que va atornillado el arduino, y ahí, resultaron los primeros dos problemas con piezas, tanto “logísticos” como problemas que tuvo la impresora al imprimir y simplemente todo se fue a la mierda.

Básicamente el primer problema fue con el ángulo al que iban a estar las patas, me explico, en el diseño que tenía de referencia, y en lo que yo consideraba ideal, era que las patas estén de la siguiente forma



Evidentemente mis habilidades artísticas en paint no demuestran demasiado bien a lo que me refiero pero creo que relativamente se entiende la idea, las patas se encontraban en un mal ángulo respecto a las esquinas de la base, las dibujó en un ángulo el cual me da mucho menos rango de movimiento, pero no logré hacer que entienda muy bien eso y las terminó imprimiendo.

El segundo problema fue meramente culpa de la impresora, no fue nada que realmente se pudiera prever, son cosas que a veces pasan, este problema fue que la maquina por alguna razón

se volvio loca y se puso a imprimir para arriba, obviamente arruinando la pieza y por consecuente desperdiциando mucho material que podia haber sido aprovechado.

Finalmente el ultimo problema que hubo fue que en el cuadrado en el que se encastra el servo para que tenga un movimiento de rotacióen de derecha a izquierda, estabamos con el mono sacando las medidas de los servos y el arduino, por ende hicimos ejemplos tanto del arduino como de los servos, y la verdad esto fue un problema que podia haber sido previsto por los dos pero no nos dimos cuenta en ningun momento, básicamente estabamos probando que todo encastrara bien, y colocamos un cuadrado que cumpla el papel de ser un servo, para verificar que entrara bien, y al momento de terminar el plano los dos nos olvidamos de retirar el cuadrado que usamos de referencia, por lo tanto se imprimió junto a todo

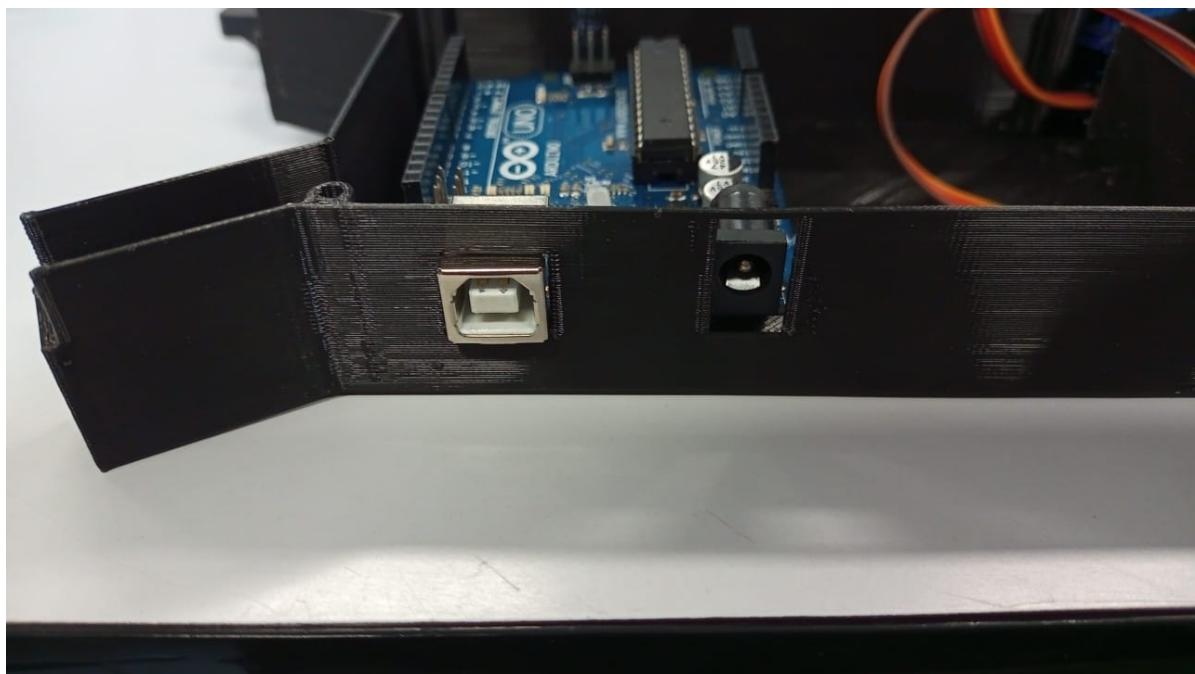


También hubo otros problemas menores que se fueron solucionando a medida que se hacían los diferentes prototipos, como por ejemplo, hubo varios problemas con la posición del arduino, tornillos, espacio dentro de la misma araña.

En cuanto al arduino, el problema fue que no consideramos que tenia que entrar además un porta baterías, una pila de 4v y tampoco tuvimos en cuenta que íbamos a tener que conectarlo la mayoría del tiempo durante el desarrollo, por lo que la solucion fue rotar el arduino, haciendo esto logramos poder colocar tanto el arduino como las pilas y creo que también todo otro componente a futuro que se necesite en este mismo.

En cuanto a los tornillos, fue difícil conseguir la posición exacta de estos en lo que respecta al arduino, el primer problema se dio con el problema del espacio, como tuvimos que rotar el arduino las muescas para los tornillos que ya habían sido colocados previamente ya no concordaban, así que se tuvo que volver a tomar las medidas correspondientes y ponerlas

Ese problema lo considero completamente arreglado, ya que por más que encaje un solo tornillo es seguro suficiente ya que tampoco se va a mover demasiado, pero con un poco de... magia y fuerza pude poner dos tornillos y ahora puedo decir con total seguridad que no se va a salir, de hecho, tener que sacarlo seria un problema ya que está demasiado encastrado. Lo cual me lleva a la siguiente modificación, ya que tuvimos que rotar el arduino, los puertos de este se estaban viendo bloqueados, y era esencial tener constante acceso a ellos durante el desarrollo de la araña, por lo que Consorti bien sugirió decidimos hacerle agujeros en donde estaban los puertos, con esto brindando una buena solución al problema



04/10/2023

El primer problema que tuve fue con el tamaño de la carcasa de la araña, ya que no tuve en cuenta todos los componentes que iban a tener que estar dentro de esta, por lo tanto tuvo que haber un rediseño de esta, haciéndole agujeros para los puertos de energía y el que se conecta a la pc, del arduino.

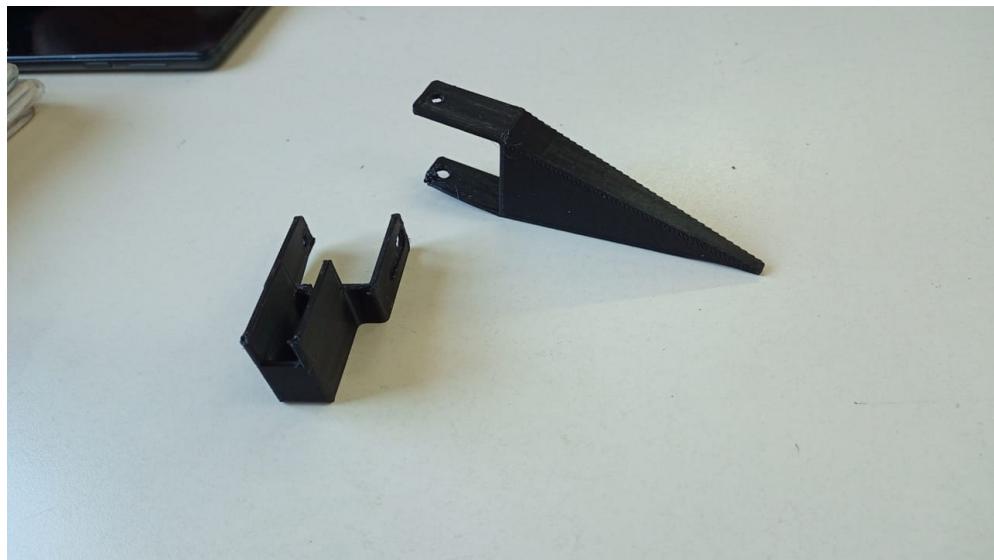


También teniendo en cuenta que la primera versión de la tapa salió demasiado mal, la impresora 3d volvió a imprimir para arriba, y cuando hicimos una que quedo bien tuvimos que volver a hacerla porque necesitábamos que sea más alta para que entraran todos los componentes, ya que no tome en cuenta todos los componentes que tenían que entrar y con el tamaño que tenía la tapa no iba a poder colocar todo

Más específicamente no tuve en cuenta el lugar que ocupaba el arduino, al lado de este se tenia que colocar un porta baterías, y arriba de estos iba a estar probablemente una protoboard con todas las conecciones de todos los componentes, iba a tener que entrar el circuito de todos los servos con su respectivo controlador y estando conectado con una batería de 9v para alimentar el arduino, y tambien estaria faltando todo lo que sería del módulo bluetooth, para poder controlar la app por el celular mediante una aplicacion de AppInventor.

Las impresiones de las piezas de las patas se están imprimiendo, sin embargo puedo avanzar simulando el código de los primeros 4 servos desde tinkercad, primero que todo hice el circuito con los componentes correspondientes, los 4 servos y una fuente para poder alimentar a los servos con 5v, también hice en una protoboard una suerte de joystick para poder simular el funcionamiento de la aplicación que vamos a hacer en appinventor, con esta aplicación tengo la idea de poder controlarlo desde cualquier celular con la aplicación, sin embargo en tinkercad no hay un módulo bluetooth así que lo tuve que conectar directamente al arduino.

05/10/2023



Las patas estaban finalmente impresas, sin embargo todo lo que podía salir mal salió mal, incluso en las cosas que ya estaban hechas ya sea por dimensiones o por falta de materiales adecuados



El primer problema que hubo fue que el aspa del servo no encajaba correctamente, por lo que me puse a limar el agujero de la pieza en un intento desesperado de poder hacer que encaje correctamente, sin embargo todo mi esfuerzo fue inutil y se tuvo que recurrir a la ayuda de Consorti, sin embargo Consorti tampoco pudo hacerlo a mano o calentando un destornillador con mi encendedor de confianza, para intentar darle forma, hasta ahora la solución definitiva para este problema fue agarrar el mini-torno conocido como Dremel



y con una broca ensanchar el dichoso agujero maltrecho. Una vez el agujero tuvo el tamaño deseado para poder situar el dichoso motor surgió un nuevo problema, las dichosas patas de araña no contaban con el tamaño milimétrico deseado, ese tamaño el cual nos permitirá encajar el azpa del servo y completar con éxito nuestro dia de trabajo no existía.

De cualquier forma si no hubiera tenido este problema tendría que haber descartado esta pieza igual ya que como se puede ver en la imagen está demasiado dañada, quedo muy finita y además está evidentemente mal impresa, al principio me dio risa y me pareció algo de lo cual no era necesario hacerse mucho drama, sin embargo, reflexionando, me di cuenta de que estos errores me costarían mucho tiempo de trabajo útil, no estoy muy seguro si ya lo conté en alguna parte de la carpeta de campo, pero, básicamente todo lo que pasó fue esto, estuve muchísimo tiempo esperando tener la pata, cuando finalmente la tuve vi el desastre que esta era, y ahí surge todo lo que dije anteriormente, el servo no encajaba, lo primero que hice fue limar a mano la circunferencia para poder intentar hacer que entre, pero obviamente esto es mucho trabajo y hay que tener en cuenta que el resultado final no es para nada el adecuado ya que la lima que estaba usando para ensanchar la circunferencia era una lima plana



No una circular, por lo que cada vez que la pasaba se notaba como se iba perdiendo la forma y empezaba a parecer un óvalo o estaba saliendo algo cuadriculado debido a la forma de la lima, por lo que lo que se me ocurrió a continuación fue calentar un encendedor y calentar algo metálico para darle forma al circulon't, Consorti me miró con la mayor cara de desaprobación / miedo al saber que llevo constantemente un encendedor conmigo a pesar de que no fumo, se lo pase e intentó darle forma calentando un encendedor pero tampoco pudo, antes de eso mi idea fue usar un cutter, ya tengo experiencia manejando ese tipo de herramientas ya que es algo que uso habitualmente en mi dia a dia para electronica o con muchos propósitos en general, pero, el problema con usar un cutter o trincheta es que no están demasiado permitidos en el establecimiento educativo en cuestión, al que pertenezco, por lo tanto no lo dude ni un segundo y fui directo a consultar si algún profesor conocido mio tenia algun cutter, y como era de esperarse la respuesta fue no, entonces, volví un rato a seguir limando hasta que sinceramente me harto porque el avance era extremadamente lento e improlijo, y por mas que no me hubiera hartado tenía que parar igual ya que estaba deformando todo, y ya ahí si recurri a Consorti, le conte todo lo sucedido y después de él mismo verificar que con la lima es un desastre y que con el encendedor a pesar de que estaba dando mas resultados tampoco era una herramienta muy adecuada o segura, se fue dramáticamente del aula y volvió con un torno de mano dremel,



con esta herramienta fue pan comido ensanchar la circunferencia de las dos piezas que estaban mal impresas, tuvo un resultado muy bueno y aún más importante, se logró rápido, pero de cualquier forma tuve que hablar con el Mono para que haga más grandes los agujeros. Un día dramático que se podría llamar desperdiciado, el cual ahora se invirtió en escuchar música a todo volumen y escribir esta carpeta de campo la cual estoy seguro que me va a dar más de una noche en vela y muchas mañanas libres desperdiciadas en tener que concurrir a este establecimiento educativo en vano, ya que no importa cuán rápido trabaje, al final el peor mal de todos va a ser que tenga o no tenga la estructura hecha no voy a ser capaz de que se mueva fluidamente.

Con esto concluye este dia de trabajo, el cual puedo describir como otro dia el cual no puedo avanzar debido a que no tengo la estructura de la araña y en tinkercad no me puedo hacer la idea de cómo sería el movimiento de esta, sin embargo, estuve haciendo en tinkercad un poco lo que creo que va a ser el código que haga a la araña moverse, mientras yo me encargo del

movimiento de las patas le voy a delegar a mi compañero la minuciosa tarea de hacer la app para poder controlar al robot desde cualquier teléfono.

Sin embargo a pesar de todo esto fueron avances necesarios, que si no tenía ahora los iba a tener probablemente más tarde.



(resultado final de los dichosos agujeros)

27/10/2023

Después de un poco más de 20 días sin hacer nada del proyecto me digne a empezar a hacer el código, siendo que no tengo las patas de la araña y me está suponiendo un retraso muy importante, estuve buscando algunos ejemplos de códigos que pudiera hacer, sin embargo todos los ejemplos que encontré usan 3 servos por pata, y yo únicamente dispongo de 2 servos por pata, por lo tanto tengo que adaptar los códigos según mis necesidades, pero el tema ahora es

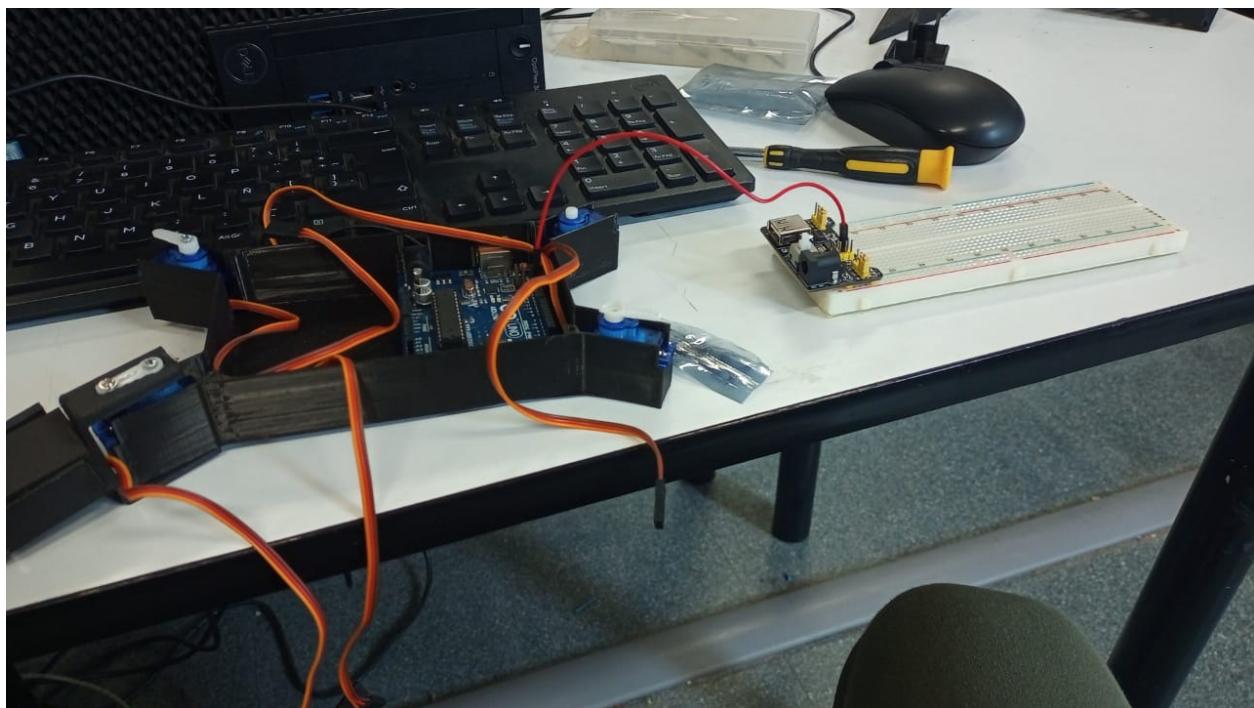
entender los códigos y que estoy haciendo, por ahora tengo en mente estos dos códigos de ejemplo.

El código que tengo en mente es [tetraPod](#) de JoaMT, sin embargo por lo primero intento ver el primer.

Luego de haber leído ese quilombo de código, llegue a la conclusión de que iba a ser mejor idea crear el mío propio

12/11/2023

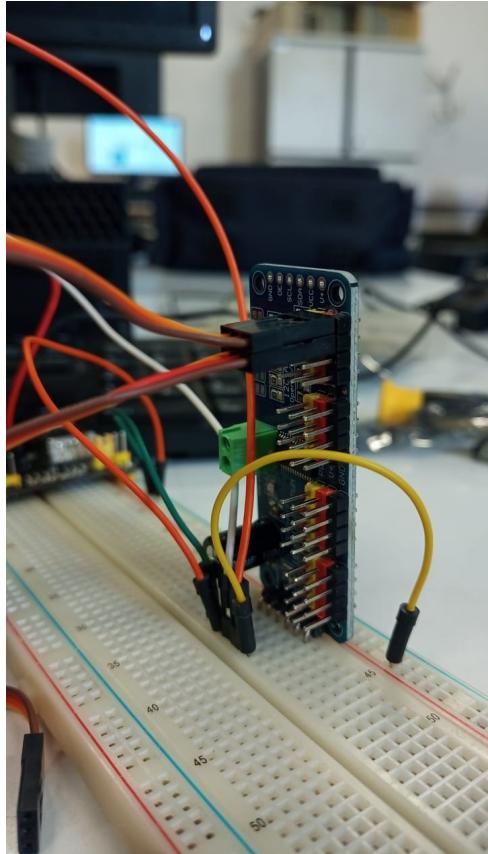
Despues de muchisimos intentos y de que el mono no entienda cosas por fin hay una pata que parece que va a funcionar, le di la tarea de intentar hacer el codigo a mi compañero Ezequiel, sin embargo luego de que no me diera ninguna señal de vida tomé la decisión de encabezar solo tanto el código como el circuito, estoy seguro de que no voy a ser capaz de conseguir un movimiento fluido con la araña, pero con que solamente se mueva para y no se caiga lo voy a considerar como logrado, he de pensar qué haré con la app y el módulo bluetooth, no creo ser capaz de llegar con la aplicación, teniendo en cuenta el limitado tiempo que me queda, sin embargo estuve haciendo unos avances en cuanto al circuito, pero no he sido capaz de probarlo ya que no dispongo de las pilas (es todo culpa de Nahuel profe, desaprobado a el)



Actualmente dispongo de dos patas funcionales, si es que el nuevo diseño del mono funciona correctamente lo cual creo que sí hará, pero para el momento de la foto todavía no estaba impreso, bueno, en la foto podemos apreciar como estuve haciendo el circuito de la araña, estuve conectando los motores y ensamblando la única pata que tenía de la araña, coloque todos los elementos en su respectivo lugar.

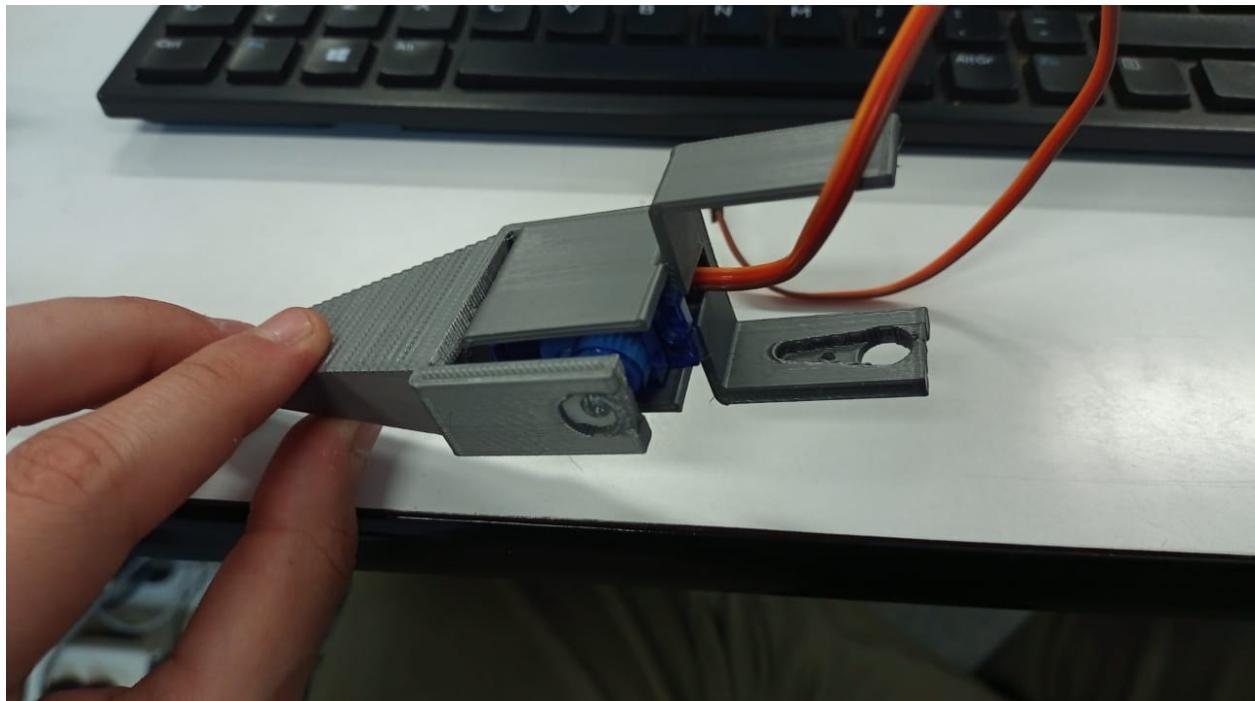
Teniendo la esperanza de poder probar aunque sea el primer servo motor de cada pata, coloque los servos y también encastre el arduino y le coloque dos tornillos, en cuanto a lo que vemos de la protoboard estuve usando un módulo para poder conectar varios servomotores y ahorrar pines, sin embargo no pude probar nada ya que no tenía las pilas, por lo que el profe me dio una fuente para motores la cual no tengo idea de como funciona, no llegue a usarla, de hecho ni pude cargar código porque tenía unas preguntas y el profe no estaba en el aula, si, así es, te estoy culpando consorti. En cuanto al código, estoy pensando en hacer una función para el

movimiento en cada dirección, y que cuando se presione el botón en la app llame a esa función y se mueva, sin embargo todavía no tengo mucha idea del funcionamiento ni de los ángulos en los que tengo que mover las patas, francamente los ángulos es lo que más me preocupa, va a ser difícil de calcular bien y tengo miedo de romper algún servo así si es que se puede



17/11/2023

Hoy cuando ya por fin tenia las nuevas patas que iba a usar resulta que no agarran bien a la partecita blanca del servo, obviamente todo esto porque el mono quiso poner la muesca del lado de adentro y no de afuera como siempre estuvo



, por lo tanto, necesito decirle que haga esa pieza otra vez y para que ya funcione y pueda tener todas las patas, me quedan 2 semanas, estoy muy atrasado, no creo llegar ni a 20 páginas de carpeta de campo bien hechas

21/11/2023

Creo que ya va siendo hora de avanzar el código, primero que todo lo que hay que pensar es como lo adaptaria para que funcione con la aplicación de appinventor, por mas que no vaya a terminar bien el codigo creo poder hacer la app, sin embargo no creo ni poder probarlo ya que todavía me falta una pata, y dudo poder coordinar bien con el Mono para hasta ese entonces,

pero, si no lo avanza voy a desaprobar, así que prefiero empezar a hacerlo por más que no vaya a quedar todo como quería.

Creo ya haberlo explicado brevemente más arriba, pero mi idea es estructurar el código de la siguiente forma, quiero que haya una función para cada dirección de movimiento, por ejemplo, void move UP etc, con esto podría hacer que la aplicación en appinventor llame a estas funciones según los botones que presione el usuario presione, sin embargo como menciona, el movimiento de las patas debe ser independiente del movimiento de la parte superior, que es donde se supone estaría el brazo, he de aclarar que debido a el tiempo con el que dispongo esta funcionalidad no va a ser agregada, y menos mal porque probablemente iba a ser horrible, retomando, he de coordinar cuidadosamente los movimientos de las patas para intentar lograr el efecto deseado.

Como ya mencione una de las cosas que más me atterra acerca del desarrollo es definir los ángulos de rotación adecuados para que cada pata se mueva en la dirección deseada, ya que experimentar con diferentes valores y observar cómo afecta a los movimientos de araña además de que me va a tomar mucho tiempo que ya no tengo, tengo miedo de dañar los servos

En cuanto a la app de appinventor estaba pensando en hacer una simple cruceta ya que no va a tener mucha más funcionalidad, la cruceta obviamente sería para representar cada dirección de movimiento

```
void moveForward() {  
    servo1.write(anguloServo1_Forward);  
    servo2.write(anguloServo2_Forward);
```

```
}
```

```
void moveBackward() {  
    servo1.write(anguloServo1_Backward);  
    servo2.write(anguloServo2_Backward);  
}
```

24/11/2023

Estuve trabajando en la lógica del código, a pesar de no tener todavía los ángulos correspondientes de cada pata tengo la lógica que tiene que seguir solo me falta aplicarle los ángulos correspondientes, fui con el mono y ya si todo se imprime bien estaría teniendo todas las patas de la araña, a pesar de que las tendría a una semana o menos de la entrega del proyecto es un gran avance ya que por fin voy a poder empezar a probar código

por más que no funcione, soy consciente de que me voy a llevar la materia así que tengo que adelantar el proyecto lo más posible para no tener que hacer tanto cuando esté en diciembre, en cuanto a las actualizaciones del código, agregue en el void loop lo siguiente

```
void loop() {
```

```
if (digitalRead(buttonUP) == LOW) {  
  
    moveUP();  
  
} else if (digitalRead(buttonDOWN) == LOW) {  
  
    moveDOWN();  
  
} else if (digitalRead(buttonRIGHT) == LOW) {  
  
    moveRIGHT();  
  
} else if (digitalRead(buttonLEFT) == LOW) {  
  
    moveLEFT();  
  
} else {  
  
    // Si ningún botón está presionado, regresa los servos a sus posiciones predeterminadas  
  
    updateServoPositions(defaultAngle1, defaultAngle2, defaultAngle3, defaultAngle4);  
  
}  
  
}
```

Lo que hace este código de Arduino es que se ejecuta dentro de una función llamada `loop()`. Esta función se ejecuta de manera continua, repitiendo el contenido una y otra vez.

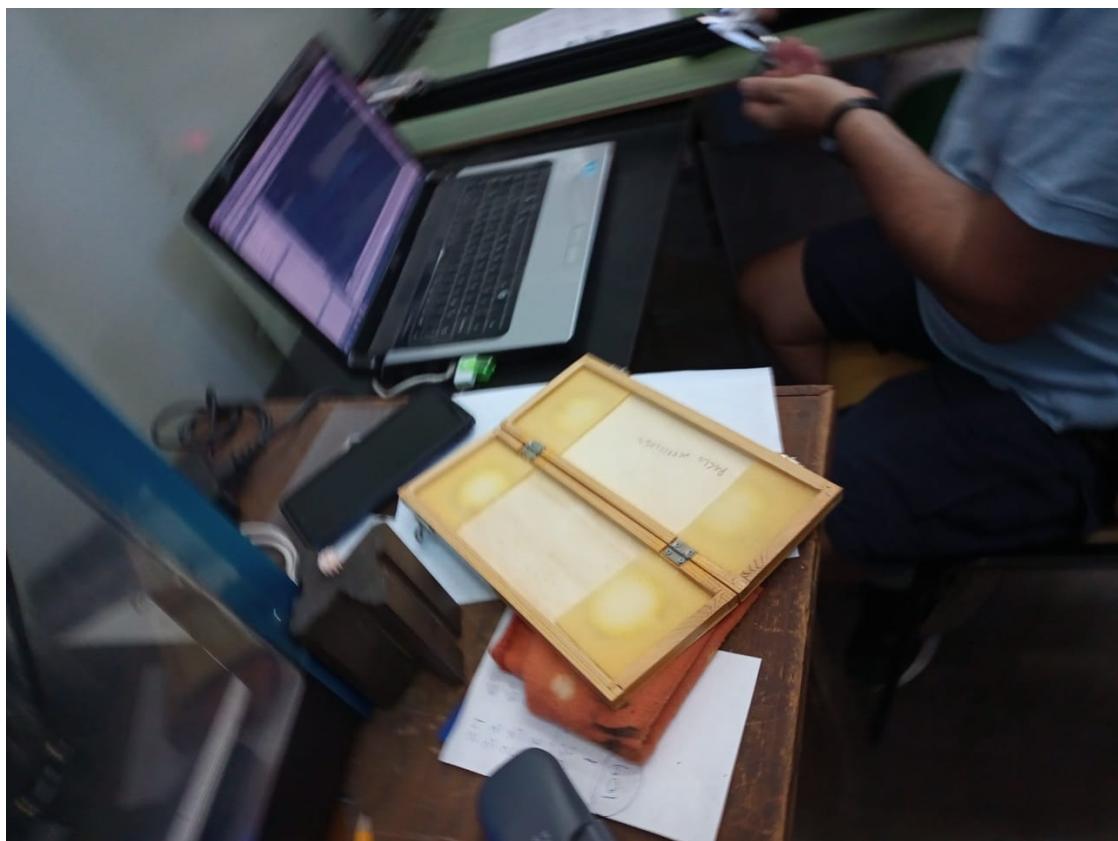
El código verifica el estado de varios botones utilizando la función `digitalRead()`. Cada botón está asociado a una dirección específica (arriba, abajo, derecha, izquierda). Si el botón está presionado (LOW), se llama a la función correspondiente para mover el dispositivo en esa dirección. Cada una de estas funciones (`moveUP()`, `moveDOWN()`, `moveRIGHT()`, `moveLEFT()`) probablemente contenga la lógica para controlar los servomotores y realizar el movimiento deseado

Si ninguno de los botones está presionado, se llama a la función updateServoPositions() con los ángulos predeterminados (defaultAngle1, defaultAngle2, defaultAngle3, defaultAngle4). Esta función probablemente restablece los servomotores a sus posiciones iniciales o predeterminadas.

En conclusión el código controla el movimiento de un dispositivo utilizando cuatro botones asociados a las direcciones arriba, abajo, derecha e izquierda. Cuando se presiona un botón, se llama a la función correspondiente para mover el dispositivo en esa dirección. Si ningún botón está presionado, se restablecen los servomotores a sus posiciones predeterminadas.

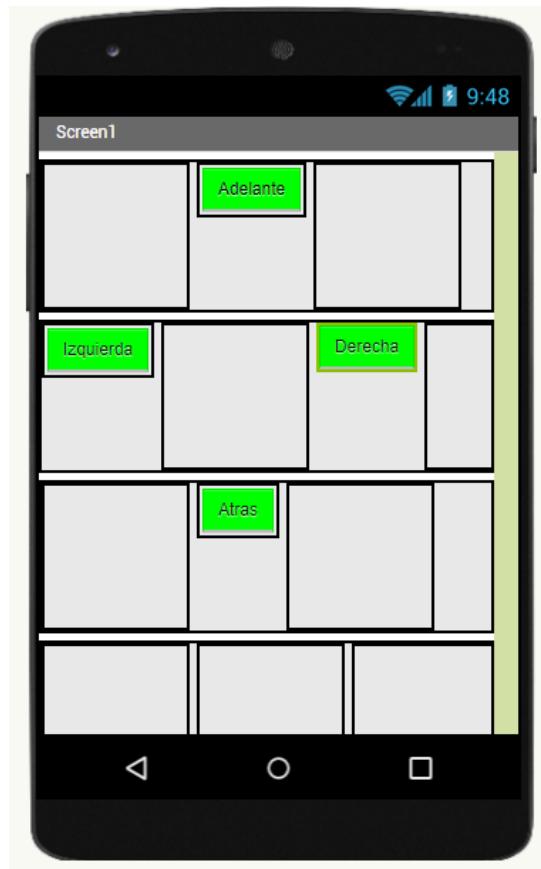
Con esta configuración, podrá controlar el robot a través de Bluetooth utilizando una aplicación personalizada en App Inventor. Cada vez que presione un botón en la aplicación, se enviará un comando al Arduino, que interpretará el comando y ajustará los ángulos de los servos en consecuencia, logrando así el movimiento deseado del robot.

Retomando el tema, por fin pude volver a coordinar con el mono para finalizar la impresión de las patas, puede que no termine de hacer el código para la entrega final, pero ya voy a tener todas las patas para intentar a prueba y error, por mas que estoy intentando imitar el movimiento de las patas de la araña desde tinkercad no estoy logrando mucho y lo poco que logré estoy seguro que no va a hacer que la araña camina bien o que siquiera se mantenga de pie



En cuanto a la aplicacion de celular ya fueron hechos dos modelos, los cuales no los probe ya que no me acuerdo como hacer los correspondientes codigos para conectar el arduino con el modulo bluethoot, tenia un ejemplo sin embargo lo borre sin querer, me duele mucho pensar en eso todavia, pero es lo que hay

Este seria el prototipo de aplicacion que hizo mi companero ezequiel, que probablemente tampoco funciona el codigo si es que lo tiene, por lo tanto tengo que averiguar sobre eso



Estuve teniendo avances con el código, a pesar de que te puedo garantizar que intenta desesperadamente hacer que chatgpt me lo pueda hacer no logre nada, intente simular el movimiento que tendrían que hacer las patas al avanzar lo mejor posible, sin embargo no voy a estar seguro hasta que no lo pruebe en la vida real, sin mencionar que desde tinkercad solo estoy simulando el movimiento de un solo servo por pata y no los dos por pata como debería estar haciendo

```
void moveUP() {  
    servo1.write(90);
```

```
delay(100);

servo4.write(45);

delay(500);

servo3.write(30);

delay(300);

servo2.write(90);

delay(500);

servo1.write(90);

delay(300);

servo4.write(0);

delay(500);

updateServoPositions(defaultAngle1, defaultAngle2, defaultAngle3, defaultAngle4);

}
```

La función moveUP() en el código controla los ángulos de rotación del primer servo de cada pata,. Sin embargo, en una configuración típica de patas de araña con dos servos por pata, cada servo tiene una función específica en el movimiento de la pata.

En una pata típica con dos servos:

El primer servo (por ejemplo, servo1) controla el movimiento de la pata de izquierda a derecha. El segundo servo (por ejemplo, servo2 conectado al servo1) controla el movimiento del resto de la pata de arriba a abajo.

En el código, moveUP() solo actualiza los ángulos de servo1 y servo4, dejando los ángulos de servo2 y servo3 sin cambios

Esto podría causar problemas en la secuencia de movimientos de la araña, ya que no estás considerando el movimiento completo de la pata.

Si quiero realizar un movimiento hacia adelante completo en una pata de araña, deberías tener en cuenta la secuencia completa de movimientos para ambos servos en esa pata

Creo no haber hablado antes hacerca de la funcion para restablecer la posicion de cada pata, hice este agregado debido a las siguientes razones

```
void updateServoPositions(int angle1, int angle2, int angle3, int angle4) {  
    servo1.write(angle1);  
    servo2.write(angle2);  
    servo3.write(angle3);  
    servo4.write(angle4);  
    delay(500); // Añade un pequeño retraso para dar tiempo a los servos a alcanzar las  
    posiciones  
}
```

Creo que laa función para restablecer la posición de los servos es crucial para garantizar un comportamiento controlado y predecible en el robot araña. En el código, esta función se llama updateServoPositions() y se utiliza para volver a la posición predeterminada de los servos

cuando no se está presionando ningún botón en la aplicación.

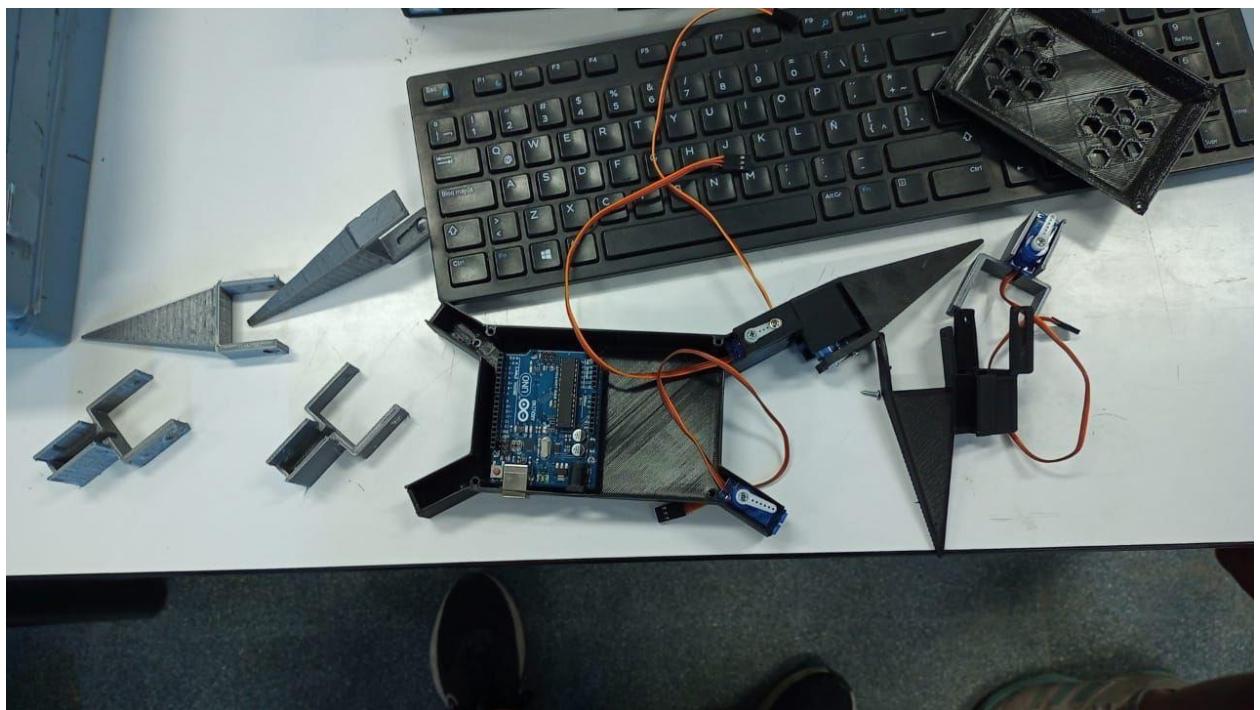
La utilidad de esta función radica en varios aspectos:

Posicionamiento inicial: Cuando se prende el robot araña o cuando no estás dando comandos específicos, es esencial que los servos regresen a una posición inicial conocida y estable. Esto ayuda a evitar movimientos inesperados o comportamientos no deseados.

Seguridad: Al restablecer las posiciones de los servos cuando no están en modo de movimiento, se reduce el riesgo de dañar el mecanismo o los servos debido a movimientos no controlados. Esto es especialmente importante durante la fase de desarrollo y prueba.

Inicio de movimiento: Cuando el usuario presiona un botón para iniciar un movimiento, la función `updateServoPositions()` garantiza que los servos estén en una posición controlada antes de ejecutar un nuevo comando. Esto contribuye a una transición más suave entre diferentes movimientos y evita movimientos bruscos.

En increíbles e inesperadas noticias de último momento, conseguí por fin tener todas las patas, a pesar de que tienen imperfecciones menores no es nada que no se solucione con un tornillo o dos



Esto significa que por fin voy a poder probarlo, sin embargo me queda exactamente una semana para hacerlo, por lo que temo no poder, este fin de semana voy a trabajar lo mas posible en el codigo para poder al menos hacerlo caminar

sea o no con la aplicacion, esa araña va a caminar, por mas que parezca tuerta, y si no lo logro, tampoco pasa nada, puedo venir en diciembre y trabajar mas tranquilo, cualquiera de las dos opciones son aceptables, que sea lo que dios y la patria quieran