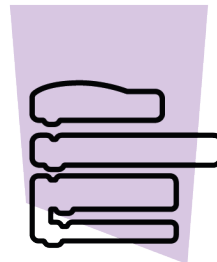
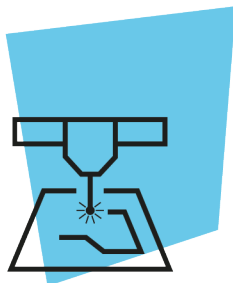
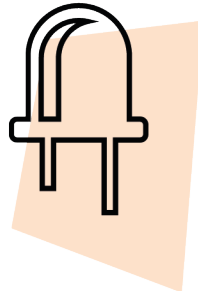
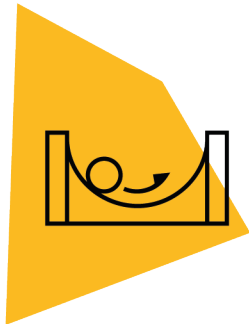
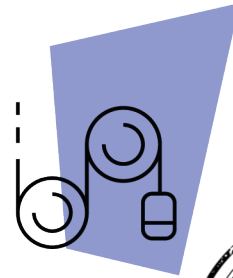


ROBÓTICA

Módulo 2



Acelerômetro e Giroscópio

AULA 27

GOVERNADOR DO ESTADO DO PARANÁ

Carlos Massa Ratinho Júnior

SECRETÁRIO DE ESTADO DA EDUCAÇÃO

Renato Feder

DIRETOR DE TECNOLOGIA E INOVAÇÃO

Andre Gustavo Souza Garbosa

COORDENADOR DE TECNOLOGIAS EDUCACIONAIS

Marcelo Gasparin

Produção de Conteúdo

Michelle dos Santos

Cleiton Rosa

Validação de Conteúdo

Cleiton Rosa

Revisão Textual

Adilson Carlos Batista

Projeto Gráfico e Diagramação

Edna do Rocio Becker

Ilustração

Jocelin Vianna

2021



Este trabalho está licenciado com uma Licença Creative Commons
Atribuição NãoComercial - CompartilhaIgual 4.0 Internacionall

Aula 01
O que já vimos?

Aula 02
Arduino:
Bibliotecas
e Funções

Aula 03
Código Morse

Aula 04
Semáforo Inteligente
com IR

Aula 05
Semáforo Completo
com Display

Aula 06
Matriz de LED 8X8

Aula 07
Desenhando na matriz de LEDs

Aula 08
Painel de Senhas

Aula 09
Escrevendo mensagens

Aula 10
Robô Autônomo

Aula 11
Sensor de Chuva

Aula 12
Sensor de Umidade
do Solo

Aula 13
Irrigador Automático

Aula 14
Feedbacks + Inventário I

Aula 15
Teclado Matricial de Membrana

Aula 16
Servos Motores

Aula 17
Fechadura Eletrônica

Aula 18
Controlando
Servos Motores

Aula 19
JoyStick Shield

Aula 20
Braço Robótico

Aula 21
Sensor de Movimento Presença

Aula 22
Sensor de Som

Aula 23
Sensor de Umidade e
Temperatura

Aula 24
Termômetro Digital

Aula 25
Sensor de Gás e Fumaça

Aula 26
Acelerômetro e
Giroscópio

Aula 27
Motor de Passo

Aula 28
Receptor IR e Controle Remoto

Aula 29
Módulo RF 433mhz - I

Aula 30
Relé

Aula 31
Módulo RF 433mhz - II

Aula 32
Projeto CHAT via RF

Aula 33
Módulo Wireless

Aula 34
Comunicação do Módulo WI-FI
em HTML

Aula 35
Módulo WI-FI - IoT
com Sensores

Aula 36
Módulo WI-FI - IoT
com Atuadores (LED)

Aula 37
Monitor de Sensores em HTML I

Aula 38
Monitor de Sensores
em HTML II

Aula 39
Mostra de Robótica

Aula 40
Feedbacks + Inventário III

Aula 26
Sensor de Gás
e Fumaça

Aula 27
Acelerômetro
e Giroscópio

Aula 28
Motor de Passo

Sumário

Introdução	2
Objetivos desta Aula	2
Competências Gerais Previstas na BNCC	3
Habilidades do Século XXI a Serem Desenvolvidas	4
Lista de Materiais	4
Roteiro da aula	5
1. Contextualização	5
2. Montagem e Programação	9
3. Feedback e Finalização	18
Videotutorial	19



Introdução

Um dos maiores consoles da atualidade é a captura de movimentos. Com isso tem mudado a forma com que temos interagido com os dispositivos eletrônicos. Você já deve ter utilizado esta funcionalidade em seu smartphone, conforme a tela se ajusta a posição da imagem ou com games em comandos através de gestos para jogos. Já se perguntou como isso funciona?

Nesta aula, estudaremos o Acelerômetro e Giroscópio. Esses sensores nos permitem medir a direção e a aceleração de objetos em relação à gravidade, indicando a posição.



Objetivos desta Aula

- Conhecer o Módulo Acelerômetro e Giroscópio MCU 6050;
- Realizar a conexão entre o Módulo e o Arduino;
- Coletar e visualizar os dados fornecidos pelo Módulo através da programação;
- Identificar potencialidades oferecidas pelo Módulo na robótica.



Competências Gerais Previstas na BNCC

[CG02] - Exercitar a curiosidade intelectual e recorrer à abordagem própria das ciências, incluindo a investigação, a reflexão, a análise crítica, a imaginação e a criatividade, para investigar causas, elaborar e testar hipóteses, formular e resolver problemas e criar soluções (inclusive tecnológicas) com base nos conhecimentos das diferentes áreas.

[CG04] - Utilizar diferentes linguagens – verbal (oral ou visual-motora, como Libras, e escrita), corporal, visual, sonora e digital –, bem como conhecimentos das linguagens artística, matemática e científica, para se expressar e partilhar informações, experiências, ideias e sentimentos em diferentes contextos e produzir sentidos que levem ao entendimento mútuo.

[CG05] - Compreender, utilizar e criar tecnologias digitais de informação e comunicação de forma crítica, significativa, reflexiva e ética nas diversas práticas sociais (incluindo as escolares) para se comunicar, acessar e disseminar informações, produzir conhecimentos, resolver problemas e exercer protagonismo e autoria na vida pessoal e coletiva.

[CG09] - Exercitar a empatia, o diálogo, a resolução de conflitos e a cooperação, fazendo-se respeitar e promovendo o respeito ao outro e aos direitos humanos, com acolhimento e valorização da diversidade de indivíduos e de grupos sociais, seus saberes, identidades, culturas e potencialidades, sem preconceitos de qualquer natureza.

[CG10] - Agir pessoal e coletivamente com autonomia, responsabilidade, flexibilidade, resiliência e determinação, tomando decisões com base em princípios éticos, democráticos, inclusivos, sustentáveis e solidários.



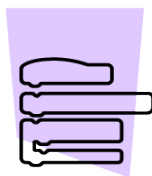
Habilidades do Século XXI a Serem Desenvolvidas

- Pensamento crítico;
- Afinidade digital;
- Resiliência;
- Resolução de problemas;
- Colaboração;
- Comunicação.



Lista de Materiais

- 1 placa Arduino Uno R3;
- 1 cabo USB;
- 4 jumpers fêmea-fêmea;
- 1 Módulo Acelerômetro e Giroscópio MPU6050;
- 1 placa Sensor Shield V5.0;
- 3 servos motores (SG90 ou MG90S);
- Notebook;
- Software Arduino IDE.



Roteiro da Aula

1. Contextualização (15min):

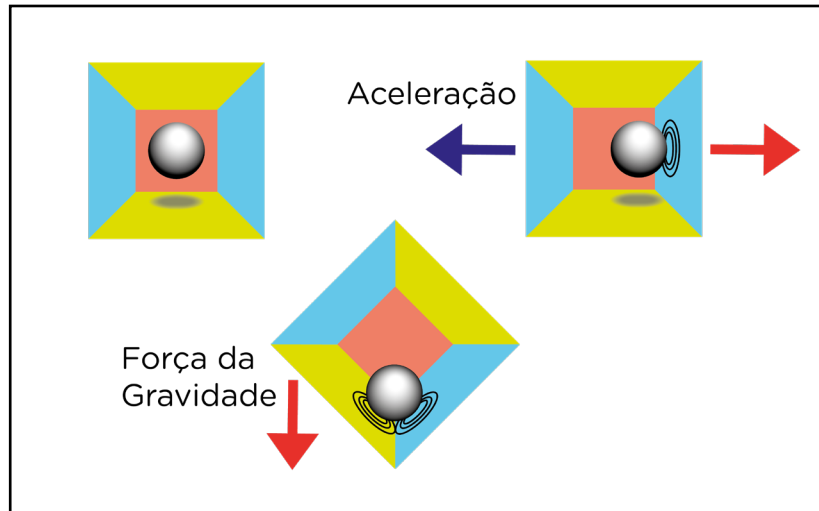
Diversos estudos têm sido desenvolvidos sobre captação e reconhecimento de movimento, entre elas as atividades humanas. Um dos mais conhecidos é o posicionamento automático da imagem, conforme o movimento em smartphones, tablets, câmeras digitais, reprodução de vídeos em 360 graus e capturas panoramas ou imagens 360.

No entretenimento, é usado em jogos, com a identificação do movimento humano, dispensando o uso de controles e joysticks. Outra finalidade é monitorar a execução de exercícios físicos, fornecendo dados como intensidade e duração da atividade, calculando a distância e percurso; e na segurança, em câmeras de monitoramento capazes de detectar movimentos, servindo-se da tecnologia de reconhecimento de imagens.

Encontrado em aplicativos que simulam o movimento e volantes de carros, realidade aumentada e virtual, quadricópteros, estabilidade de imagem óptica e eletrônica, entre outros.

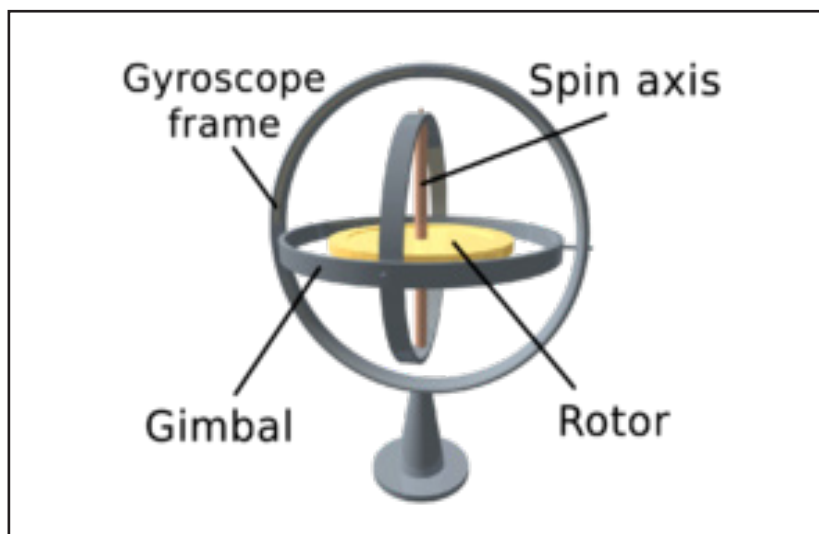
O Acelerômetro possui paredes sensíveis ao toque, então quando o sensor é deslocado de cima para baixo, ou da esquerda para direita, graças à força exercida sobre ele, é percebida a direção de deslocamento. Medindo a posição do corpo através da gravidade, conforme figura 1. Tem a vantagem de ser pequeno, barato e com boa sensibilidade, sendo utilizado em sistemas nos quais pode detectar, mensurar ou captar qualquer movimento ou vibração.

Figura 1 - Funcionamento interno de um Acelerômetro



Com adição do Giroscópio as medições de movimentos e mudanças de posição tornam-se mais precisas graças à adição de novas dimensões de movimento, como a rotação ou voltas do dispositivo, utilizando da força da gravidade para indicar a posição de um determinado objeto no espaço, podendo identificar se você gira algo em seu próprio eixo ou saber se está apontando para cima ou para baixo, conforme figura 2.

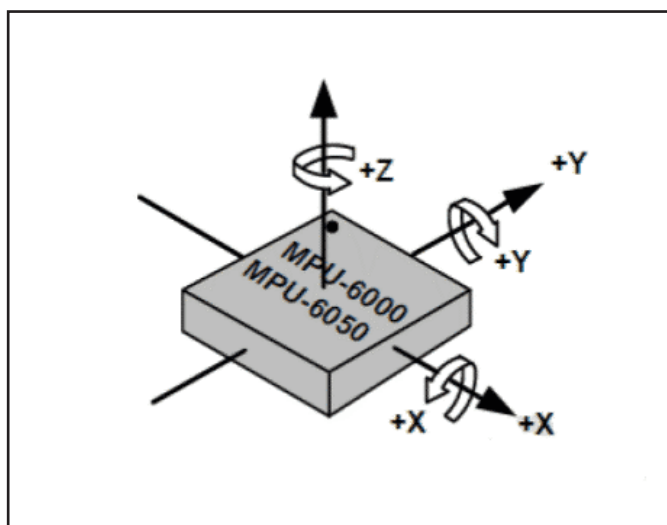
Figura 2 - Giroscópio



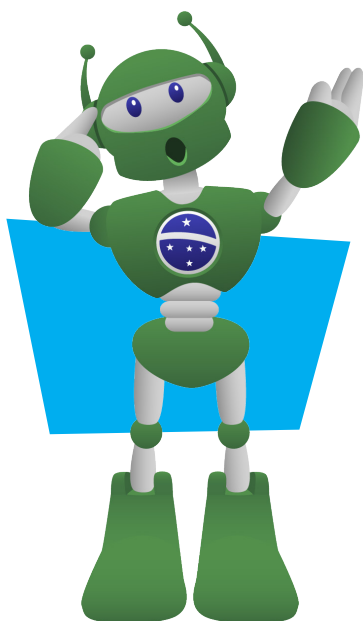
Fonte: <https://pt.wikipedia.org/wiki/Girosc%C3%B3pio>

O Módulo MPU6050 apresenta dois recursos na mesma placa, ou seja, contém em um único chip um Acelerômetro e um Giroscópio. No total são 6 eixos, sendo três para o Acelerômetro e 3 para o Giroscópio, que capturam os valores x, y e z ao mesmo tempo, conforme figura 3.

Figura 3 - Módulo MPU6050



Fonte: <https://capsistema.com.br/index.php/2020/11/16/mpu-6050-gy-521-tutorial-arduino/>



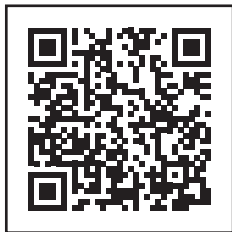
Imagine utilizar o celular como uma raquete virtual, com o desenvolvimento deste projeto poderá aumentar as possibilidades de aplicação que pode fazer no cotidiano, na interação do usuário não só com o computador, mas com outros dispositivos oferecendo maiores recursos. Esse campo de estudo é muito promissor para os próximos inventos.





Para Saber Mais...

Conheça o primeiro celular a ficar conhecido com o uso do Giroscópio.



<https://pt.ifixit.com/Teardown/iPhone+4+Gyroscope+Teardown/3156>

A reportagem “Vigilância digital: aplicativo transforma celular em monitor de segurança”, comenta sobre o dispositivo especializado em detectar sons e movimentos suspeitos em um cômodo da casa ou em um quarto de hotel. Confira!

<https://www.techtudo.com.br/noticias/2017/12/vigilancia-digital-aplicativo-transforma-celular-em-monitor-de-seguranca.ghtml>



A reportagem “Quantas informações sensíveis um relógio inteligente revela sobre você?”, comenta sobre relógios inteligentes (smartwatches) que monitoram atividades físicas e podem fornecer informações sigilosas dos usuários. Confira!

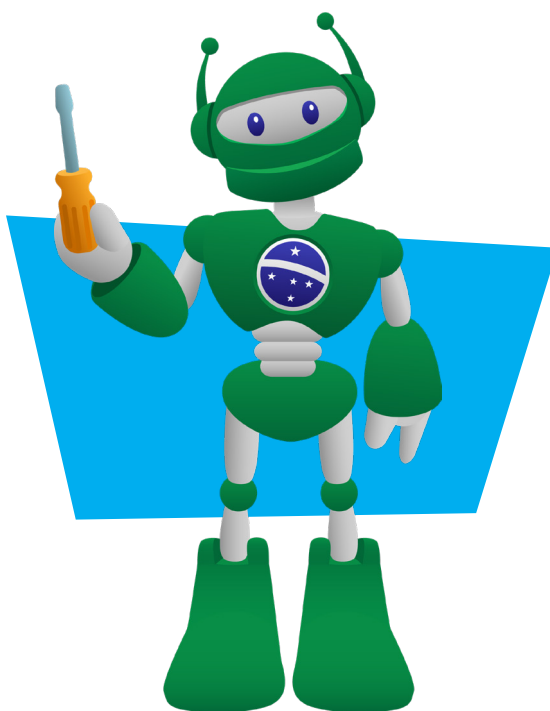
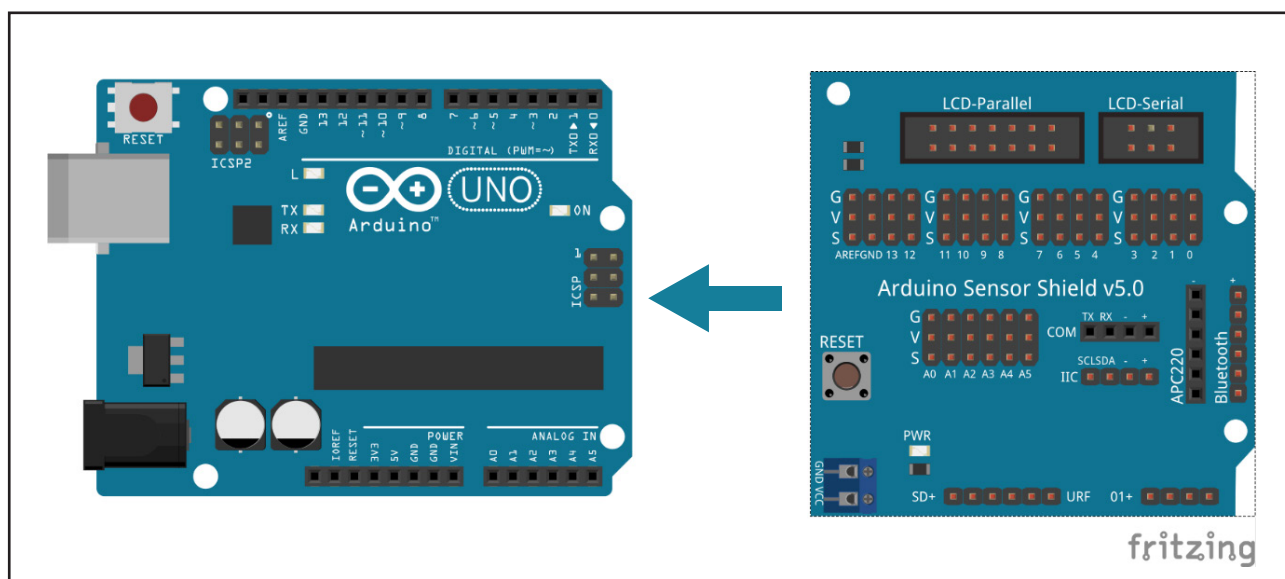


<https://itforum.com.br/quantas-informacoes-sensiveis-um-relogio-inteligente-revela-sobre-voce/>

2. Montagem e Programação (60min):

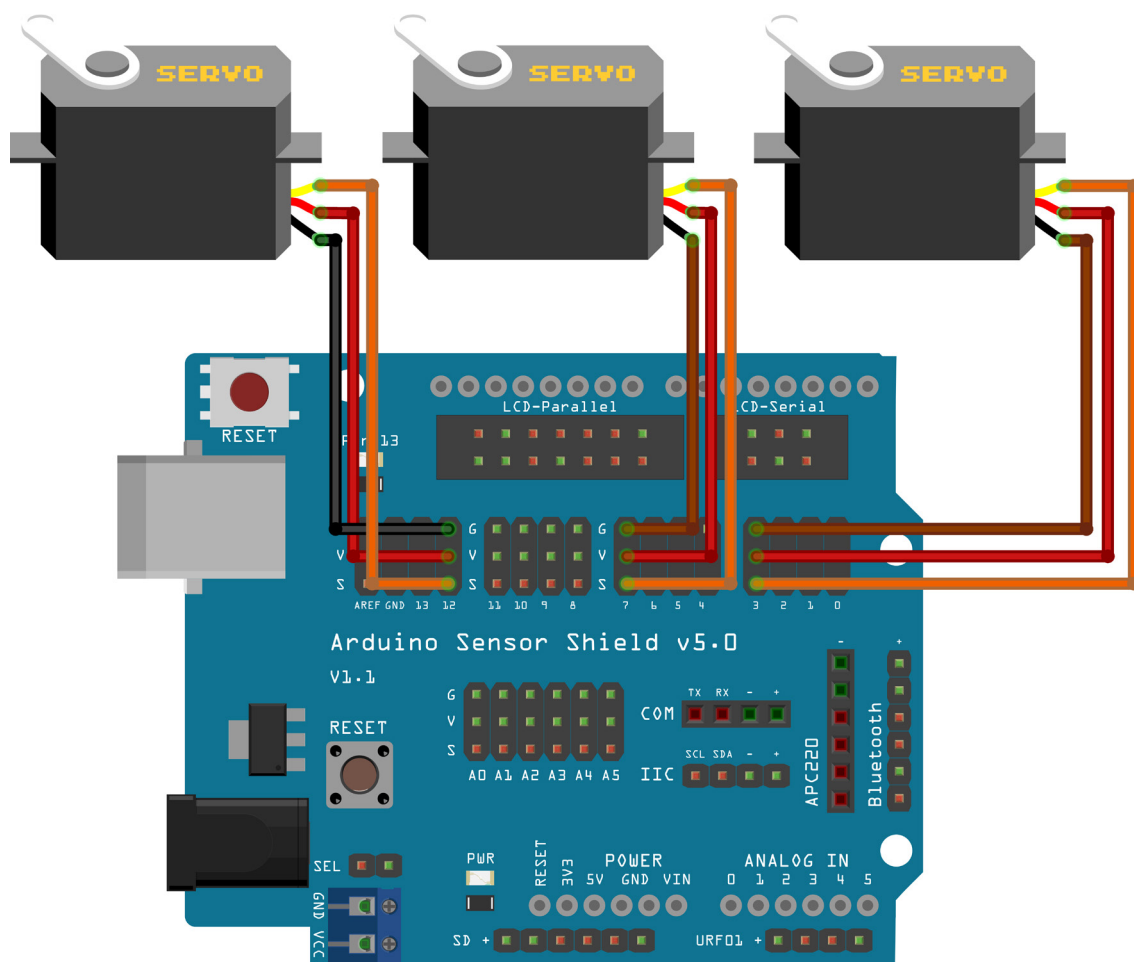
Antes de iniciar a montagem, acople a placa Sensor Shield sobre a placa Arduino, como mostra a Figura 4.

Figura 4 - Acoplando a placa Sensor Shield sobre o Arduino



Conecte os três servos motores nas portas digitais 3, 7 e 12 do sensor shield respeitando as linhas: G (fios marrons), V (fios vermelhos) e S (fios laranjas), como indicado na Figura 5.

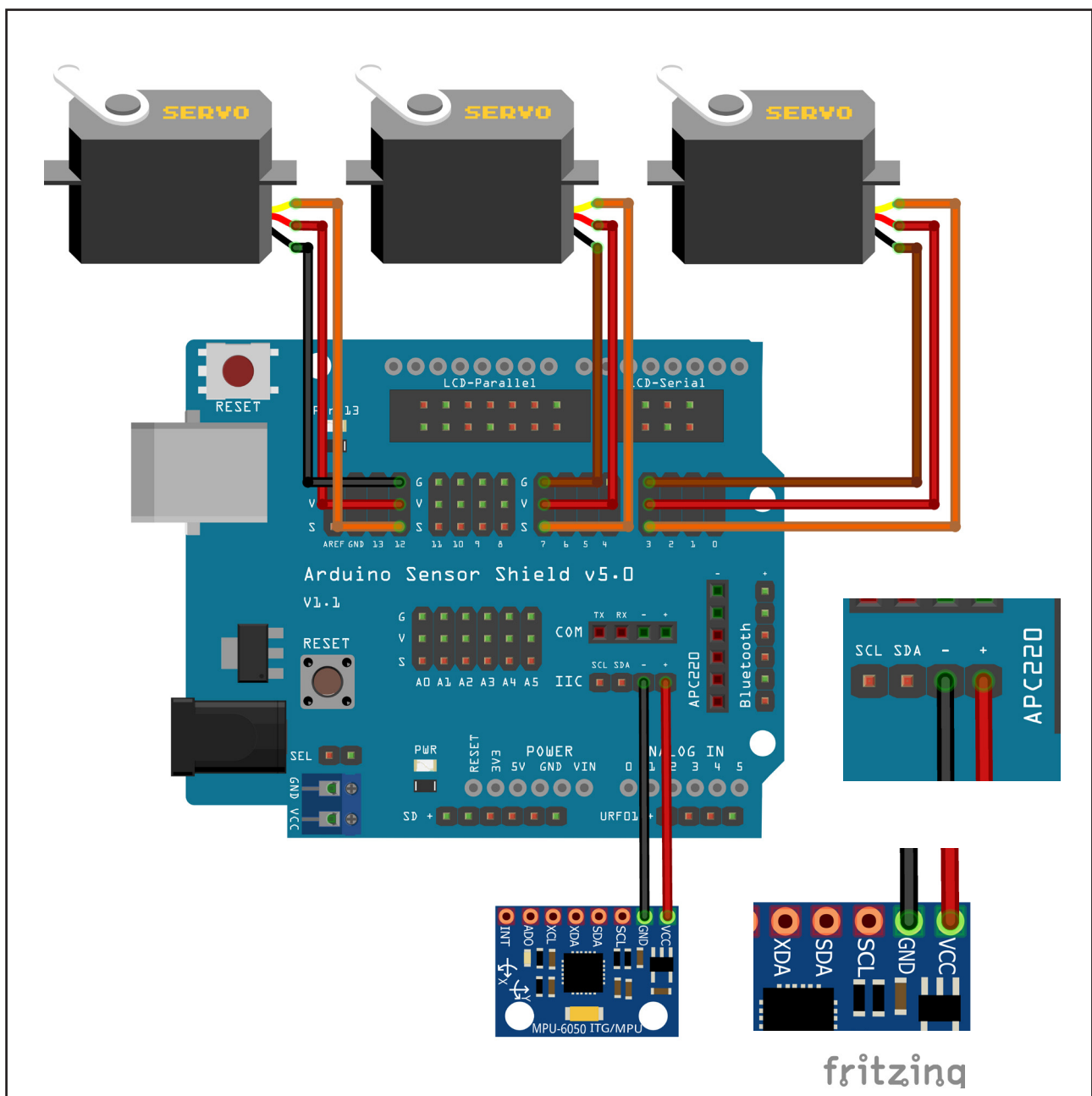
Figura 5 - Conexão dos Servos Motores na placa Sensor Shield



fritzing

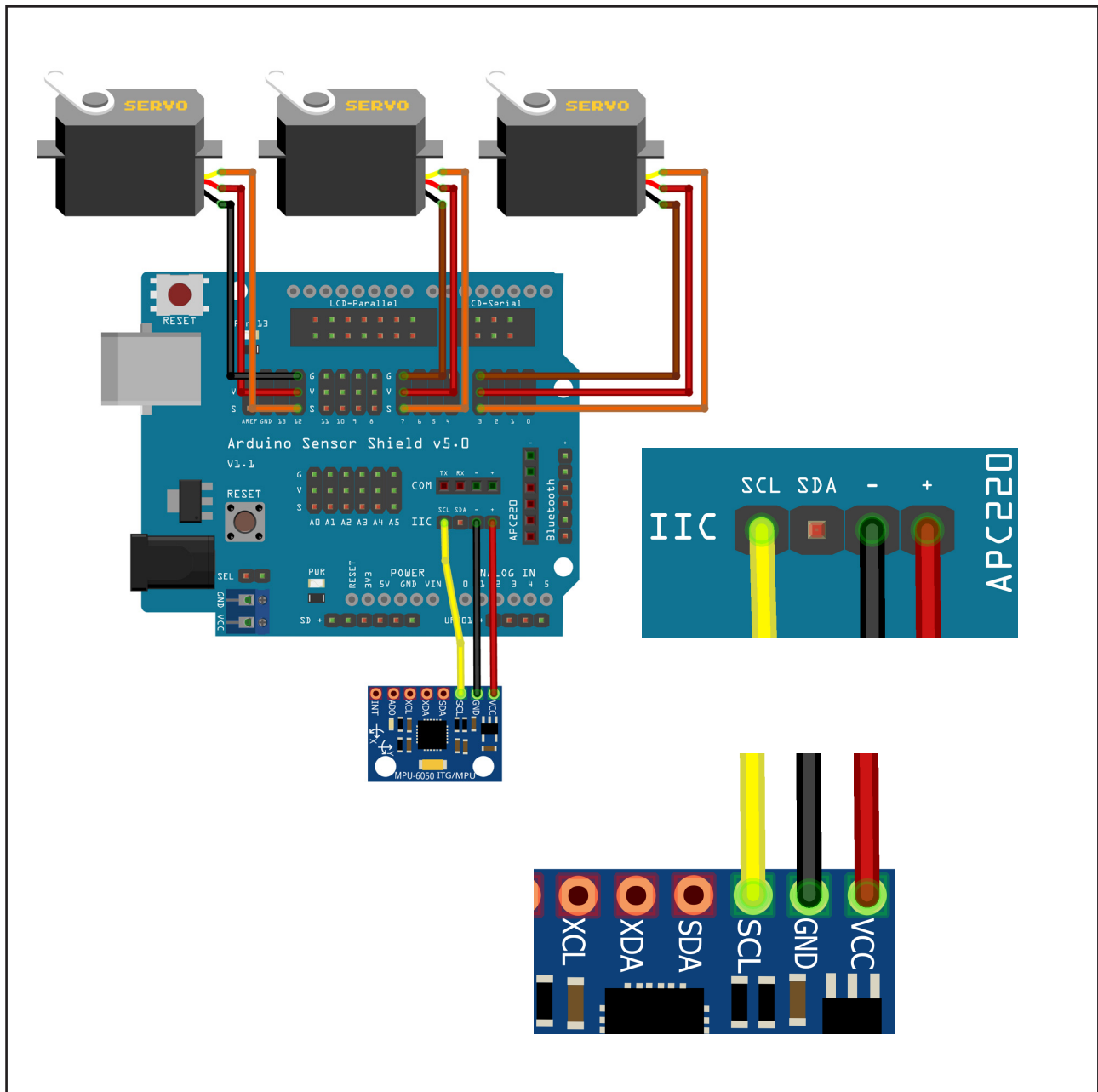
Utilize dois Jumpers para alimentar o Módulo Acelerômetro e Giroscópio, conectando os pinos **VCC** e **GND** aos pinos + e - da placa Sensor Shield, como indicado na Figura 6.

Figura 6 - Alimentação do Módulo Acelerômetro e Giroscópio



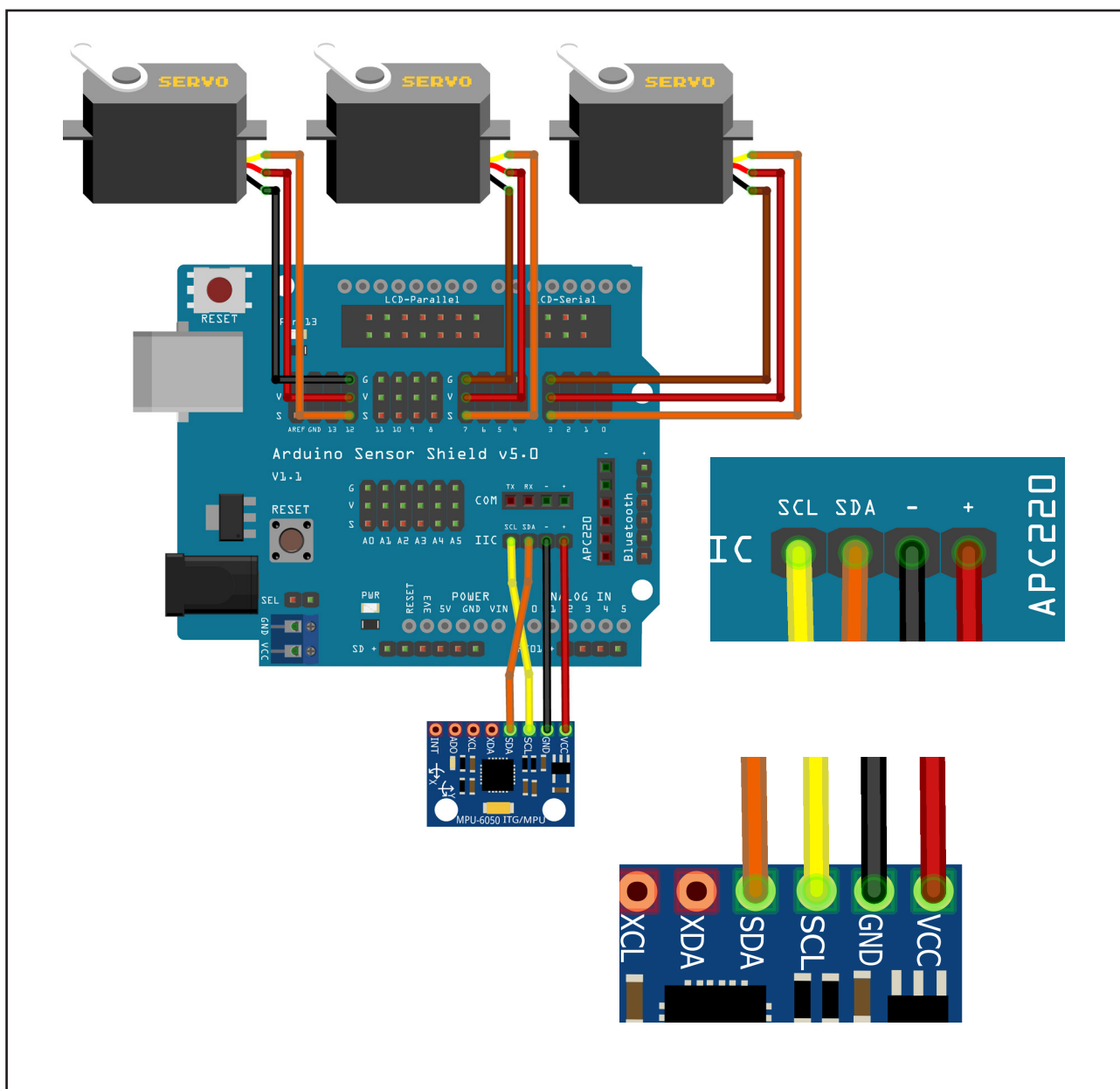
Utilize outro Jumper para conectar o pino **SCL** do Módulo ao pino **SCL** ou **A5** do Sensor Shield, como indicado na Figura 7.

Figura 7 - Conexão dos Servos Motores na placa Sensor Shield



Para finalizar a montagem, conecte através de outro Jumper, o pino **SDA** do Módulo ao pino **SDA** ou **A4** do Sensor Shield, como indicado na Figura 8.

Figura 8 - Conexão dos Servos Motores na placa Sensor Shield





Agora, vamos programar!

Com os componentes eletrônicos montados, vamos programar, por codificação, o funcionamento do Módulo Acelerômetro e Giroscópio.

Linguagem de programação por código

Para iniciar a programação, conecte a placa Arduino ao computador, através de um cabo USB, para que ocorra a comunicação entre a placa microcontroladora e o software Arduino IDE.

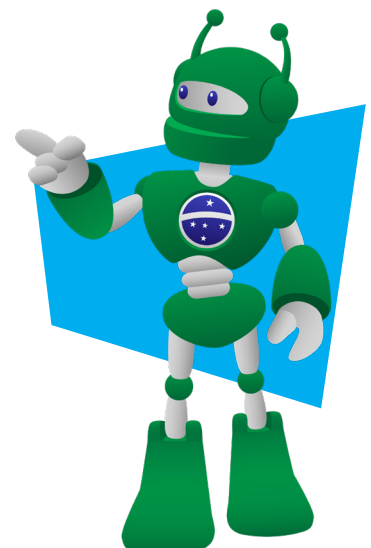
Nessa programação, utilizaremos as bibliotecas “**Servo**”, “**Wire**” (instaladas por padrão no Software Arduino IDE) e a biblioteca “**MPU6050** tockn” que auxiliará no controle do módulo acelerômetro e giroscópio (para recordar como instalar uma biblioteca, consulte a **Aula 05 - Softwares Arduino IDE e mBlock** do **módulo 1**).

<<https://github.com/fred-corp/GY6050/archive/refs/heads/master.zip>>

No software IDE, crie um sketch e lembre-se de selecionar a porta que o computador atribuiu ao Arduino; então, digite ou copie e cole o código-fonte de programação, conforme apresentado no quadro 1:

Atenção!

Ao copiar o código diretamente do pdf, evite quebra da página (e consequentemente erros na compilação), copiando o código por partes.



Quadro 1 - Código-fonte da programação na linguagem do Arduino (Wiring)

```

/*****
/* Aula 27 - Acelerômetro e giroscópio */
/* Programação do módulo acelerômetro e giroscópio MPU6050 */
/* que visa controlar três servomotores através da */
/* rotação do módulo nas direções dos eixos X, Y e Z. */
/* Também é impresso no monitor serial os valores dos */
/* ângulos e temperatura do sensor. */
/*****
/* Link para download da biblioteca MPU6050 tockn. */
/* https://github.com/Tockn/MPU6050\_tockn */
/*****
/* Inclui as bibliotecas necessárias. */
#include <MPU6050_tockn.h>
#include <Servo.h>
#include <Wire.h>
/* Cria os objetos de controle para cada servomotor. */
Servo servoX;
Servo servoY;
Servo servoZ;
/* Cria o objeto de controle do módulo MPU6050. */
MPU6050 mpu6050(Wire);

void setup() {
  /* Inicia a comunicação serial com velocidade de 9600. */
  Serial.begin(9600);
  /* Inicia a comunicação Wire(protocolo I2C). */
  Wire.begin();
  /* Inicia o objeto de controle do módulo MPU6050. */
  mpu6050.begin();
  /* Calibração automática do módulo MPU6050. */
  /* O parâmetro true mostra a calibração no monitor serial.*/
  mpu6050.calcGyroOffsets(true);
  /* Atribui as portas de controle para cada servomotor. */
  servoX.attach(3);
  servoY.attach(7);
  servoZ.attach(12);
}

void loop() {
  /* Atualiza a posição do módulo MPU6050. */
  mpu6050.update();
  /* Coleta os dados do módulo e armazena nas variáveis. */
  int eixoX = mpu6050.getAngleX();

```

```

int eixoY = mpu6050.getAngleY();
int eixoZ = mpu6050.getAngleZ();
float temp = mpu6050.getTemp();

/* Mapeia os dados de -90 a 90 para valores de 0 a 180. */
int Xmapeado = map(eixoX, -90, 90, 0, 180);
int Ymapeado = map(eixoY, -90, 90, 0, 180);
int Zmapeado = map(eixoZ, -90, 90, 0, 180);

/* Restringe os dados para valores de 0 a 180. */
int restritoX = constrain(Xmapeado, 0, 180);
int restritoY = constrain(Ymapeado, 0, 180);
int restritoZ = constrain(Zmapeado, 0, 180);

/* Imprime os dados no monitor serial. */
Serial.print("eixo X: " + String(restritoX) + "\t");
Serial.print("eixo Y: " + String(restritoY) + "\t");
Serial.print("eixo Z: " + String(restritoZ) + "\t");
Serial.print("Temp.: " + String(temp, 2) + "°C\n");

/* Posiciona os servomotores. */
servoX.write(restritoX);
servoY.write(restritoY);
servoZ.write(restritoZ);
}
/*****
/* Obtendo outros dados do sensor MPU6050: */
/*
/* Aceleração no eixo X: mpu6050.getAccX() */
/* Aceleração no eixo Y: mpu6050.getAccY() */
/* Aceleração no eixo Z: mpu6050.getAccZ() */
/*
/* Giro no eixo X: mpu6050.getGyroX() */
/* Giro no eixo Y: mpu6050.getGyroY() */
/* Giro no eixo Z: mpu6050.getGyroZ() */
/*
/* Aceleração angular no eixo X: mpu6050.getAccAngleX() */
/* Aceleração angular no eixo Y: mpu6050.getAccAngleY() */
/* Aceleração angular no eixo Z: mpu6050.getAccAngleZ() */
/*
/* Giro angular no eixo X: mpu6050.getGyroAngleX() */
/* Giro angular no eixo Y: mpu6050.getGyroAngleY() */
/* Giro angular no eixo Z: mpu6050.getGyroAngleZ() */
/*
*****/

```

A seguir, compile o programa pressionando o botão **Verificar** (botão com sinal de tique) para verificar se não há erros de sintaxe. Estando o código correto, o próximo passo é realizar a transferência do programa para o Arduino. Pressione o botão **Carregar** (botão com uma seta apontando para a direita), para realizar upload do programa para o Arduino

Após a transferência do programa para o Arduino, ao girar o módulo acelerômetro e giroscópio nas direções X, Y e Z, você terá o controle do movimento dos três servomotores. .



Desafio:

Que tal utilizar o Acelerômetro e Giroscópio para controlar o braço robótico? Adapte a programação de modo a atender aos requisitos para o controle através deste Módulo!



E se...?

O projeto não funcionar, se atente a alguns dos possíveis erros:

- a. Verifique se os Jumpers estão nos pinos certos, se estão na mesma coluna dos terminais dos componentes, fazendo assim as conexões;
- b. Verifique se a programação está adequada a cada porta digital.

3. Feedback e Finalização (15min):

a. Confira, compartilhando seu projeto com os demais colegas, se o objetivo foi alcançado para o funcionamento do Giroscópio e Acelerômetro.

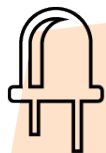
b. Analise seu projeto desenvolvido, de modo a atender aos requisitos para funcionamento do Módulo Giroscópio e Acelerômetro.

c. Reflita se as seguintes situações ocorreram:

i. Colaboração e Cooperação: você e os membros de sua equipe interagiram entre si, compartilhando ideias que promoveram a aprendizagem e o desenvolvimento deste projeto?

ii. Pensamento Crítico e Resolução de Problemas: você conseguiu identificar os problemas, analisar informações e tomar decisões de modo a contribuir para o projeto desenvolvido?

d. Reúna todos os componentes utilizados nesta aula e os organize novamente, junto aos demais, no kit de robótica.



Videotutorial

Com o intuito de auxiliar na montagem e na programação desta aula, apresentamos um videotutorial, disponível em:



<https://rebrand.ly/a27robotica2>

Acesse, também, pelo QRCode:

