**Bem vindo ao Manual do Robô Ardu!!!!**

Através do **Robô Ardu** você irá fazer coisas incríveis e aprender muito sobre automação com Arduino e desenvolvimento de software. O seu **Robô Ardu** foi projetado para que você aprenda estes conceitos através da possibilidade de você determinar o que **Robô Ardu**  irá fazer. Tudo isso sem a necessidade de conhecimentos aprofundados em microeletrônica e programação. Mas para conseguir programar o **Robô Ardu** você deve ler alguns conceitos e fazer os exemplos deste manual. Para fazer as atividades é só você baixar o ambiente de programação Arduino no GitHub <https://github.com/RobotArdu>.

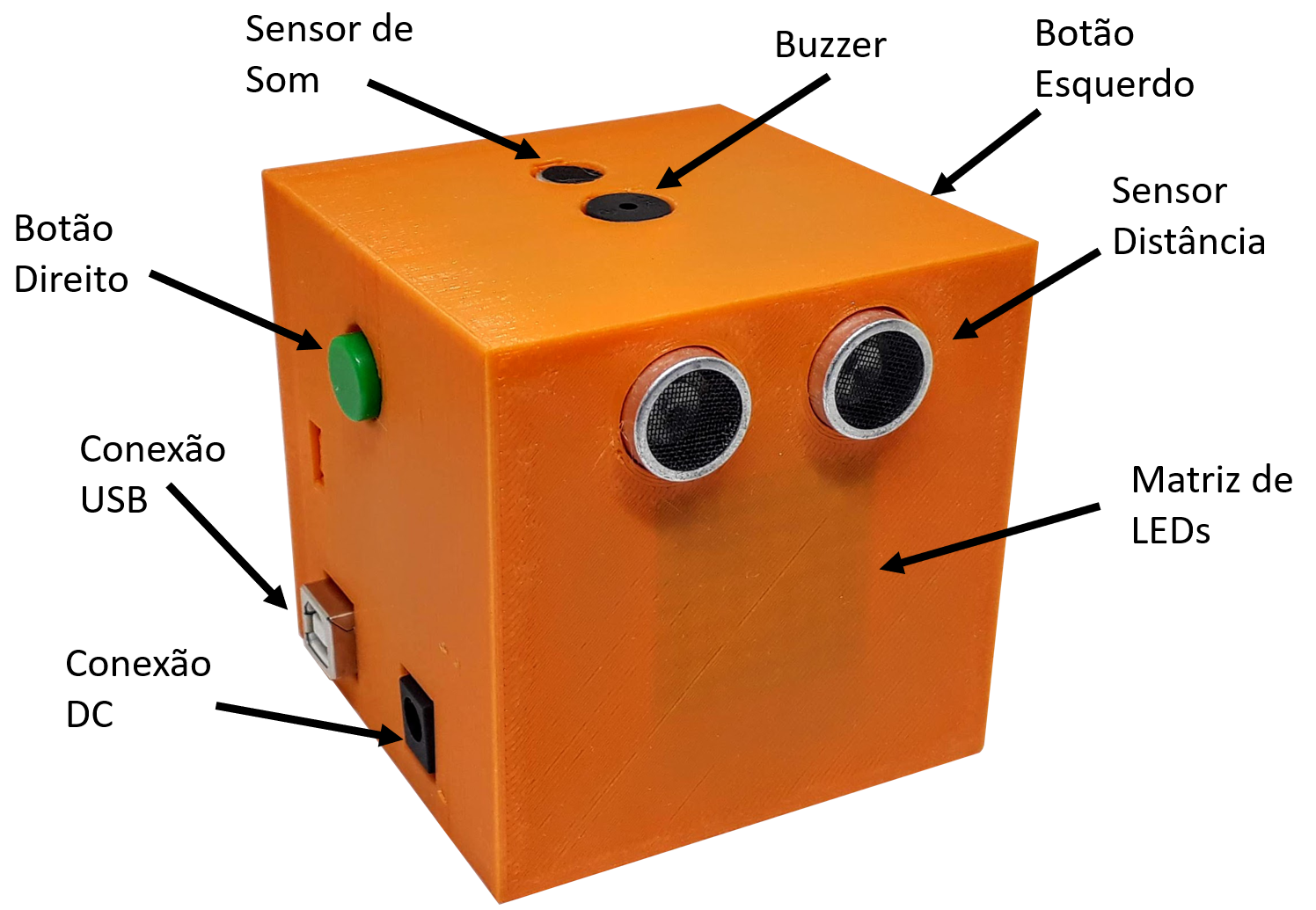
Depois de compilar o código é só instalar no **Robô Ardu** e como mágica o seu robô vai funcionar como você pensou.

Primeiro Exemplo

|  |  |
| --- | --- |
| #include <Robotica\_Cognitiva.h>  void loop() {  Sentimento.Feliz();  } |  |

**Componentes de Hardware do Robô Ardu**

|  |  |  |
| --- | --- | --- |
| COMPONENTE | DESCRIÇÃO | IMAGEM |
| ARDUINO UNO | A placa Arduino é o cérebro do robô. Ela é uma plataforma de computação que nos permite programar de inúmeras formas diferentes. | Resultado de imagem para arduino uno |
| BEEP (BUZZER) | O buzzer é um pequeno alto falante capaz de transmitir sons de variadas frequências. |  |
| SENSOR DE SOM (PALMS) | Através do sensor de som, você pode captar o som do ambiente, como palmas. |  |
| BOTÃO | É um componente que permite realizar uma ação quando o mesmo é apertado. | esultado de imagem para botao arduino  Resultado de imagem para imagem botao 12x12 |
| SENSOR ULTRASONICO | Permite detectar quando objetos se aproximam. |  |
| MATRIZ LED | Componente com um grupo de led’s que em conjunto podem ser ligados para forma uma figura ou uma letra ou números. |  |



**Comandos de Programação basicos do Arduino**

|  |  |  |
| --- | --- | --- |
| **ITEM** | **DESCRIÇÃO** | **COMANDO** |
| Estrutura | No Arduino a função setup() é chamada no momento em que o programa começa. Essa função é executada somente uma vez, quando o Arduino é iniciado ou quando é resetado. | **void setup() {**    **}** |
| Estrutura | A função loop() faz precisamente o que seu nome indica: ela repete-se continuamente permitindo que seu programa funcione dinamicamente. | **void loop() {**    **}** |
| Variáveis | Podemos associar um nome a um tipo de dado utilizando variáveis para guardarmos valores que queiramos usar depois em outra parte do nosso código. Você pode mudar o valor contido na sua variável quantas vezes quiser ao longo do programa. | int nota = 5;  float salario = 0.578;  bool aceso = True;  char letra = ‘m’; |
| Constantes | Constantes são variáveis que recebem o valor apenas uma vez e não podem ser alteradas ao longo do programa.Usamos constantes para facilitar a leitura do código. | int const media = 7; |
| Condição (if/else) | Funciona exatamente como o se e senão da língua portuguesa, só executando algo caso a condição seja verdadeira. Exemplos Literais:  Se tiver almoço então comerei.  Nesse exemplo fica claro que só comerei se tiver almoço.  Se tiver almoço, vou comer. Senão vou cozinhar alguma coisa.  O senão na segunda sequência indica que tenho uma resposta tanto para o caso de ter almoço quanto para o caso de não ter. | **if (condição){**  **o que acontece caso a condição seja verdadeira**  **}else{**  **o que acontece caso a condição seja falsa**  **}** |
| Interação (while) | Traduzindo para o português, while significa enquanto. Ele cria um loop dentro do programa até que alguma condição seja atingida. Exemplo Literal:  Enquanto o ônibus não chega, aguardo. Dai embarco. | **while(condição){**  **o que acontece até a condição ser atingida**  **}** |
| Interação (for) | Assim como o while, o for cria um loop no programa. A diferença é a forma em que o loop é estruturado.  Primeiro ocorre a **inicialização** que acontece apenas uma vez. A cada repetição do loop, a **condição** é testada; se é verdadeira, o bloco de comandos, e o **incremento** são executados. Quando a **condição** se torna falsa, o loop termina. | **for (inicialização,condição,incremento){**  **o que acontece enquanto condição é verdadeira**  **}** |

Para maior aprofundamento nas funções utilizadas pelo arduino recomendamos o site <https://www.arduino.cc/reference/pt/>

Comandos de Programação basicos da biblioteca **Robotica\_Cognitiva**

|  |  |  |
| --- | --- | --- |
| **ITEM** | **DESCRIÇÂO** | **COMANDO** |
| Sentimento | Com este comando você poderá fazer com que **Robô Ardu** apresente na matriz de Led expressões de sentimento como Feliz e Triste. | Sentimento.Feliz();  Sentimento.Triste(); |
| Estado | O comando ***Estado*** agrupa funcionalidades para controlar eventos do **Robô Ardu.** Por exemplo, com o **Estado**, você pode verificar se um objeto se aproximou de **Ardu** ou verificar se som foi tocado e também captar os click dos seus botões. | if(Estado.Sensor\_Som()){  }  if (Estado.Sensor\_Distancia()<15){ *// O valor 15 representa uma distância de 3 cm em média de proximidade de um objeto*  }  Estado.Sensor\_Som();  Estado.Botao\_Direito();  Estado.Botao\_Esquerdo();    Estado.Sensor\_Distancia(); |
| Acao | Com este comando você poderá realizar um conjunto de ações que o **Ardu** sabe fazer. Por exemplo: ligar o display, tocar um som, imprimir uma frase no display, etc. | Acao. Apaga\_Tudo(); *// Apaga todos os Led´s do display*  Acao. Liga\_Tudo(); *// Liga todos os Led´s do display*  Acao.Imprimir(M); *// Apresenta a letra* ***M*** *no display*  Acao.Animacao(onda, 180); *// Apresenta uma animação de* ***onda*** *no display*  Acao.Imprime\_Frase(“Meu nome é Ardu”,180); *// Apresenta a frase* ***Meu nome é Ardu*** *no display*  Acao.Toca\_Buzzer(Super\_Feliz*); //Toca um som no Buzzer Super\_Feliz*  Acao.Joga\_Dado(); *//Joga um dado digital no display do robô parando em um número aleatório* |
| Sons | Conjunto de Sons que você pode utilizar na Acao.Toca\_Buzzer(...).  Exemplo: Acao.Toca\_Buzzer(Super\_Feliz*);//Toca um som no Buzzer Super\_Feliz* | Conectado  Desconectado  Botao\_Apertado  Modo\_1  Modo\_2  Modo\_3  Surpresa  OhOoh  OhOoh\_2  Carinhoso  Sono  Feliz\_1  Super\_Feliz  Feliz\_2  Triste\_1  Confuso  Pum\_1  Pum\_2  Pum\_3 |
| Caras | Conjunto de Imagens que você pode utilizar na Acao.Imprimir(...);  Exemplo:  Acao.Imprimir(raio); | smile1  smile2  smile3  coracao  surpresa  raio  dinheiro  hashtag  zero, um, ... , nove  A, B, C, ... , Z  menor\_que  maior\_que  igual  interrogacao  arroba |

1. **Usando o botão esquerdo e direito para fazer um abajur (if)**

|  |  |
| --- | --- |
| #include <Robotica\_Cognitiva.h>  void loop() {  if (Estado.Botao\_Direito()){  Acao.Liga\_Tudo();  }  if (Estado.Botao\_Esquerdo()){  Acao.Apaga\_Tudo();  }  } | Primeiro passo foi importar a biblioteca do **Robô Ardu**.  Depois dentro da função *loop()* foram usadas duas condições. Se o botão direito foi apertado o display será ligado completamente.  A outra condição é se o botão esquerdo foi pressionado. Neste caso o display do **Robô Ardu** será apagado. |

**2. Usando o botão direito para bricar de jogar dado (if)**

|  |  |
| --- | --- |
| #include <Robotica\_Cognitiva.h>  void loop() {  if (Estado.Botao\_Direito()){  Acao.Joga\_Dado();  delay(3000);  }  } | Primeiro passo foi importar a biblioteca do **Robô Ardu**.  Depois dentro da função *loop()* foi usada uma condição. Se o botão direito foi apertado será sorteado um número entre 1 e 6 e será apresentado no display do **Robô Ardu**. |

**3. Usando o sensor de som para tocar um Som de Surpresa (if)**

|  |  |
| --- | --- |
| #include <Robotica\_Cognitiva.h>  void loop() {  if (Estado.Sensor\_Som()){  Acao.Toca\_Buzzer(Surpresa);  delay(2000);  }  } | Primeiro passo foi importar a biblioteca do **Robô Ardu**.  Depois dentro da função *loop()* foi implementado um comando se o sensor de som foi acionado. Caso isso aconteça será tocado um som de surpresa. |

|  |
| --- |
| Tente você agora  Faça um abajur acionado através de palmas. Programe também para ao apertar o botão direito desligar o display do **Robô Ardu.** |

**4. Fazendo o seu Robô Ardu contar até 3 (três) com o acionamento do botão esquerdo (if)**

|  |  |
| --- | --- |
| #include "Robotica\_Cognitiva.h"  void loop() {  if (Estado.Botao\_Esquerdo()){  Acao.Imprimir(um);  delay(2000);  Acao.Imprimir(dois);  delay(2000);  Acao.Imprimir(tres);  delay(2000);  Acao.Apaga\_Tudo();  }  } | Primeiro passo foi importar a biblioteca do **Robô Ardu**.  Depois dentro da função *loop()* foi implementada uma condição se o botão esquerdo foi apertado. Caso positivo usamos a Acao de Imprimir o número 1 (*Acao.Imprimir(um);*) e programamos para ficar aparecendo por 2 segundo no display (*delay (2000);*). Fazemos isso para o número 2 e 3 na sequência e por fim limpamos o display ( *Acao.Apaga\_Tudo();*) |

**5. Fazendo o seu Robô Ardu acionar o Sentimento Feliz depois de três clicks no botão direito. (variável)**

|  |  |
| --- | --- |
| #include <Robotica\_Cognitiva.h>  int x = 0;  void loop(){  if ( Estado.Botao\_Direito()){  x=x+1;  Acao.Toca\_Buzzer(Botao\_Apertado );  }  if (x==3){  Sentimento.Feliz();  delay(2000);  Acao.Apaga\_Tudo();  x=0;  }  } | Primeiro passo foi incluir a biblioteca do **Robô Ardu**.  Depois dentro da função *loop()* foi implementada uma condição se o botão direito foi apertado. Caso positivo usamos a Acao Toca\_Buzzer (Botao\_Apertado) e somamos 1 ao valor de x. Após apertado o botão direito 3 vezes, o valor de x será 3, o que tornará a segunda condicional do programa fazendo que o robô use Sentimento.Feliz() por 2 segundos e depois apague a matriz de led e mude o valor de x para 0. |

**6. . Fazendo o seu Robo Ardu contar até 3 (três) usando o comando *for* com o acionamento do botão esquerdo (for)**

|  |  |
| --- | --- |
| #include <Robotica\_Cognitiva.h>  void loop() {  if (Estado.Botao\_Esquerdo()){  for (int i=1;i<=3;i++){  if (i==1){  Acao.Imprimir(um);  delay(2000);  }  if (i==2){  Acao.Imprimir(dois);  delay(2000);  }  if (i==3){  Acao.Imprimir(tres);  delay(2000);  Acao.Apaga\_Tudo();  }  }  }  } | Primeiro passo foi incluir a biblioteca do **Robô Ardu**.  Depois dentro da função *loop()* foi implementada uma condição se o botão esquerdo foi apertado. Caso positivo implementamos o comando for (*for (int i=1;i<=3;i++){}*). A cada iteração do *for* a variável i é acrescida de 1.  Com as condições verificamos se a variavel i é igual ao valor que qeremos imprimir.  Se i==1 usamos a Acao de Imprimir o número 1 (*Acao.Imprimir(um);*)  Observe que se i==3 além de imprimir o número 3 depois de 2 segundos apagamos o display. |

|  |
| --- |
| Tente você agora  Faça um programa que ao apertar o botão direito o **Robô Ardu** conte até 3 porém faça uma contagem decrescente, ou seja, 3..2..1**.** Faça sem o comando *for* e depois com o comando for (*for (int i=3;i>=1;i--){}*). |

**7. Fazendo o seu Robo Ardu acionar um alarme depois de 1 minuto (variável)**

|  |  |
| --- | --- |
| #include <Robotica\_Cognitiva.h>  long tempoInicial = millis();  long intervalo =60000; // 1 minuto  void loop() {  long tempoAtual = millis();  long tempoDiferenca = tempoAtual-tempoInicial;  if ( tempoDiferenca >= intervalo ){  Sentimento.Feliz();  //Acao.Toca\_Buzzer(Feliz\_1 );  delay(2000);  tempoInicial = tempoAtual;  }else{  Acao.Apaga\_Tudo();  //Acao.Toca\_Buzzer(songName);  }  } | Primeiro passo foi incluir a biblioteca do **Robô Ardu**.  Depois criamos duas variáveis, uma servirá para guardar o tempo inicial e a outra guardará o tempo necessário para ligar o alarme.  Dentro da função *loop()* criamos outras duas variáveis uma que irá guardar o tempo atual sempre que o loop reiniciar e outra que guardará a diferença entre o tempo inicial e o atual.  Quando o tempoDiferença for maior ou igual ao valor definido em intervalo o robô irá acionar o alarme por 2 segundos e em seguida reiniciar o tempo inicial.  Toda vez que o tempoDiferença for menor que o valor de intervalo o robô apagará tudo. |

**8. Fazendo o seu Robo Ardu apresentar no Display o símbolo de um raio através de dupla condição. (And)**

|  |  |
| --- | --- |
| #include "Robotica\_Cognitiva.h"  void loop() {  if((Estado.Botao\_Direito()) && (Estado.Botao\_Esquerdo())){  Acao.Imprimir(raio);  }else{  Acao.Liga\_Tudo();  }  } | Primeiro passo foi incluir a biblioteca do **Robô Ardu**.  Dentro da função *loop()* testamos se o botão direito e o botão esquerdo são pressionados simultaneamente. Isto é feito através da condição *And (&&)*.  Se sim imprimimos um raio no display se não ligamos todos os led´s do display. |

**9. Fazendo o seu Robo Ardu imprimir Ardu no Display enquanto um objeto próximo. (while)**

|  |  |
| --- | --- |
| #include "Robotica\_Cognitiva.h"  void loop() {  while(Estado.Sensor\_Distancia()<15){    Acao.Imprime\_Frase(" Ardu ",80);    }  Acao.Liga\_Tudo();  } | Primeiro passo foi incluir a biblioteca do **Robô Ardu**.  Dentro da função *loop()* através do comando *while* ficaremos em um segundo loop enquanto um objeto estiver próximo do sensor ultrasonico.  Dentro do comando *while* será executada a *Acao.Imprime\_Frase(" Ardu ",80);.*  Fora do comando *while*, ou seja, se um objeto não estiver próximo o display será ligado. |

|  |
| --- |
| **Crie o seu proprio Robô Ardu e coloque no smartphone**  Crie seu próprio exemplo e salve no editor do arduino com o nome de ardu.ino. Além disso, exporte o seu código binário compilado usando o menu *skecth* do editor do arduino, gerando o arquivo ardu.ino.standard.hex. Depois é só copiar os arquivos na memória interna do celular na pasta **roboardu**. Assim você pode através do celular na opção “Meu Robô” usar a sua programação para **Ardu**. |