

# DuinoBlocks4Kids: Instruções para Instalação, Configuração e Uso

(DB4K 4.0.0)

## 1. DB4K-CS e DB4K-SA

O DuinoBlok4Kids possui duas versões, o [DB4K-CS \(Client Side\)](#) e o [DB4K-SA \(Satandalone\)](#). O DB4K-CS é multiplataforma, no entanto, exige a instalação do Python (ferramenta Livre) e de um navegador de internet, preferencialmente o Google Chrome (Navegador Gratuito). Já o DB4K-SA está disponível para Windows 32 bits e 64 Bits, e dispensa a instalação do Python e o uso do navegador. Ambas as versões possuem exatamente as mesmas funcionalidades e é importante mencionar que **ambas rodam localmente sem necessidade de acesso à Internet**. As duas versões necessitam que o usuário tenha o software Arduino instalado na sua máquina.

O ambiente de programação DB4K possui uma [versão de demonstração online](#), no entanto, não é possível, nesta versão de demonstração, enviar os programas feitos no DB4K diretamente para uma placa Arduino conectada ao computador. Porém, pode-se baixar o programa em blocos construído no ambiente, no formato “.ino” (formato dos programas Arduino) e, então, abrir este programa no software Arduino e enviá-lo para uma placa Arduino conectada ao computador.

## 2. Instruções de Instalação, Configuração e Uso

### 2.1 Instalando o Ambiente

#### 2.1.1 **DB4K-SA** (Windows 32/64)

1. Baixe e instale o **Software Arduino 1.6.8**, <https://www.arduino.cc/en/Main/OldSoftwareReleases#00xx>.
2. Baixe e descompacte o arquivo **Pacote-DB4K-SA-B4.0.0-Win64.zip** ou **Pacote-DB4K-SA-B4.0.0-Win32.zip**
3. Copie as pastas **Ultrasonic** e **Thermistor** presentes na pasta **Bibliotecas Arduino** do pacote de instalação do DB4K para a pasta **Documentos\Arduino\libraries**. (Essas pastas possuem os arquivos necessários para se utilizar o sensor de distância e de temperatura do DB4K)
4. Execute o arquivo **DB4K-SA-B4.0.0-Win64-Setup.exe** ou **DB4K-SA-B4.0.0-Win32-Setup.exe**.

Após a instalação, um ícone para execução do DB4K- SA ficará disponível na área de trabalho.

#### 2.1.2 **DB4K-CS** (Multiplataforma)

##### 2.1.2.1 **Windows**

1. Baixe e instale o Interpretador **Python 2.7.9**.
  - a. Para **Windows 64** bits instale essa versão <https://www.python.org/ftp/python/2.7.9/python-2.7.9.amd64.msi>
  - b. Para **Windows 32** bits instale essa versão <https://www.python.org/ftp/python/2.7.9/python-2.7.9.msi>
2. Baixe e instale o **Software Arduino 1.6.8**, <https://www.arduino.cc/en/Main/OldSoftwareReleases#00xx>.
3. Baixe e descompacte o arquivo **Pacote-DB4K-CS-B4.0.0-Win.zip**.
4. Copie as pastas **Ultrasonic** e **Thermistor** presentes na pasta Bibliotecas Arduino do pacote de instalação do DB4K para a pasta pasta **Meus Documentos\Arduino\libraries**. (Essas

pastas possuem os arquivos necessários para se utilizar o sensor de distância e de temperatura do DB4K)

5. Execute o arquivo **DB4K-CS-B4.0.0-Windows-Setup.exe**.

Após a instalação, **um ícone para execução do DB4K-CS ficará disponível na área de trabalho**. O DB4K irá abrir no **navegador padrão**. Sugerimos o uso do Google Chrome, mas, outros navegadores, em princípio, podem ser utilizados.

#### 2.1.2.2 **Ubuntu 16.05 / 18.05** (para outras plataformas baseadas em Linux ainda não foram realizados testes)

1. Baixe e Instale o **Software Arduino**

- Baixe o pacote de instalação para a sua plataforma (**diretamente do site Arduino**):

**Linux 32** <https://www.arduino.cc/download.php?f=/arduino-nightly-linux32.tar.xz>

**Linux 64** <https://www.arduino.cc/download.php?f=/arduino-nightly-linux64.tar.xz>

- Copie o arquivo baixado para a sua **Pasta Pessoal** do Ubuntu e descompacte
- Abra uma janela de terminal (para isso aperte as teclas **<ctrl> <alt> <t>**)
- Digite **cd arduino-nightly** e aperte a tecla **<enter>** (↵)
- Digite **./install.sh** e aperte a tecla **<enter>** (↵)

2. Baixe e Instale o **DB4K** e as **Bibliotecas dos Sensores**

- Baixe e descompacte o arquivo **Pacote-DB4K-CS-B4.4.0-Multiplataforma.zip**
- Copie as pastas **Ultrasonic** e **Thermistor** presentes na pasta **Bibliotecas Arduino** do pacote de instalação do DB4K para a pasta **libraries** localizada dentro da pasta **Arduino** localizada na sua **Pasta Pessoal**. (Essas pastas contêm as bibliotecas necessárias para se utilizar os sensores de distância e de temperatura.)
- Copie a Pasta **DB4K-CS** para a sua **Pasta Pessoal**.
- Abra uma janela de terminal (para isso aperte as teclas **<ctrl> <alt> <t>**)
- Digite **cd DB4K-CS** e aperte a tecla **<enter>** (↵)
- Digite **chmod +x install.sh** e aperte a tecla **<enter>** (↵)
- Digite **./install.sh** e aperte a tecla **<enter>** (↵)

Após a instalação, **um ícone para execução do DB4K-CS ficará disponível na área de trabalho**. O DB4K irá abrir no navegador padrão.

(No **Ubuntu 18.05**, é possível que, por questões de segurança, seja criado um atalho sem o ícone do DB4K. Clique neste atalho e, na janela que abrir, aperte a opção **<confiar e iniciar>**)

#### 2.1.2.3 **Outras Plataformas** (Ainda não testado)

1. Baixe e instale o Interpretador **Python 2.7.9**
2. Baixe e instale o **Software Arduino 1.6.8**
3. Baixe e descompacte o arquivo **Pacote-DB4K-CS-B4.4.0-Multiplataforma**
4. Copie as pastas **Ultrasonic** e **Thermistor** presentes na pasta **Bibliotecas Arduino** do pacote de instalação do DB4K para a pasta **libraries** localizada dentro da **pasta do Software Arduino**. Essas pastas contêm as bibliotecas necessárias para se utilizar os sensores de distância e de temperatura.
5. Copie a Pasta **DB4K-CS** (descompactada no item 3) para um local de sua preferência.
6. O **DB4K** é iniciado por meio do arquivo **start.py** presente na pasta **DB4K-CS**. O DB4K irá abrir no navegador padrão. Sugerimos o uso do **Google Chrome**, mas, outros navegadores, em princípio, podem ser utilizados.

## 2.2 Configurando os Pinos de Entra e Saída

O DB4K vem com os pinos de entrada e saída utilizados pelos blocos **previamente configurados para o uso com a “caixinha mágica”** ( A caixinha mágica ( [https://youtu.be/dHW5nmfMs\\_g](https://youtu.be/dHW5nmfMs_g)) faz uso de uma placa **Arduino Mega**, e os demais materiais de robótica desenvolvidos e sugeridos para uso com o DB4K, (<https://www.youtube.com/grzBots>), fazem uso de placas **Arduino Uno**. ) No entanto, **é possível utilizar o DB4K para o controle de qualquer circuito que faça uso dos sensores e atuadores contemplados pelo DB4K**. Assim, caso se deseje modificar os pinos para o controle de circuitos montados com outra configuração de pinos e de entrada e saída, os números dos pinos podem ser reconfigurados. Toda essa configuração das pinagens, evidentemente, é feita pelo professor, ela é transparente para as crianças, ao menos em um primeiro momento.

**Caso deseje modificar os números dos pinos, edite os valores da seguinte maneira:**

### 2.2.1 Modificando os pinos de entrada e saída no DB4K-SA

Os números dos pinos devem ser modificados no arquivo `DB4K-SA\ardublockly\db4k_setup.js` (Este arquivo pode ser editado utilizando-se programas como o Notepad++ ou o Bloco de Notas do Windows). (vídeo explicativo: <https://youtu.be/ZkclQNE0dEY?t=1009>)

**IMPORTANTE:** Após modificar os valores dos pinos e salvar o arquivo de configuração, ao rodar o DB4K, é necessário fazer um **"reload" da aplicação**, de maneira que essas modificações sejam refletidas no código Arduino gerado pelo ambiente. Para tanto, entre no menu <Configurações> e selecione a opção <atualizar pinos>

**Obs.**

Existem duas possibilidades de programação dos **sensores de reflexo** (leitura digital ou analógica) para modificar o código gerado pelo bloco do sensor, você deve modificar o atributo que define o tipo de leitura no arquivo `DB4K-SA\ardublockly\db4k_setup.js` e depois fazer um **“reload” da aplicação** utilizando a opção de menu <atualizar pinos>

Existem duas possibilidades de uso da **biblioteca Ultrasonic (sensor de distância)**. Caso você tenha problemas no uso da biblioteca instalada no seu computador, tente trocar a opção de biblioteca para o sensor de distância no arquivo `DB4K-SA\ardublockly\db4k_setup.js` e depois fazer um **“reload” da aplicação** utilizando a opção de menu <atualizar pinos>

### 2.2.2 Modificando os pinos de entrada e saída no DB4K-CS

Os números dos pinos devem ser modificados no arquivo `DB4K-SA\DB4K\db4k_setup.js` (Este arquivo pode ser editado utilizando-se programas como o Bloco de Notas, o Notepad++ ou o Gedit) .

**IMPORTANTE:** Após modificar os valores dos pinos e salvar o arquivo de configuração, ao rodar o DB4K é necessário fazer um **"reload" da aplicação** de maneira que essas modificações sejam refletidas no código Arduino gerado pelo ambiente. Para tanto, entre no menu <Configurações> e selecione a opção <atualizar pinos>. Caso o reload da página não seja feito e os números dos pinos não sejam atualizados, aperte ao mesmo tempo as teclas <Ctrl> e <F5>, para forçar a atualização do cache do navegador (Chrome). Caso, ainda assim a atualização não seja feita, será necessário **limpar o histórico do seu navegador** (não é preciso limpar cookies, dados de formulários e senhas)

**Obs.**

Existem duas possibilidades de programação dos **sensores de reflexo** (leitura digital ou analógica) para modificar o código gerado pelo bloco do sensor, você deve modificar o atributo que define

o tipo de leitura no arquivo DB4K-SA\ardublockly\db4k\_setup.js e depois fazer um “**reload**” da **aplicação** utilizando a opção de menu <**atualizar pinos**>

Existem duas possibilidades de uso da **biblioteca Ultrasonic (sensor de distância)**. Caso você tenha problemas no uso da biblioteca instalada no seu computador, tente trocar a opção de biblioteca para o sensor de distância no arquivo DB4K-SA\ardublockly\db4k\_setup.js e depois fazer um “**reload da aplicação**” utilizando a opção de menu <**atualizar pinos**>

### **2.2.3 Tabela com a configuração inicial dos Pinos do DB4K**

A tabela abaixo apresenta os pinos previamente configurados no DB4K.

Componente		Pino	Componente	LED	Pino	Componente	Sinal	Pino
LED	VERMELHO	9	Display_7_SEG.	A-7	24	Display_LCD	RS	31
LED	AMARELO	10	Display_7_SEG.	B-6	22	Display_LCD	RW	33
LED	VERDE	11	Display_7_SEG.	C-4	27	Display_LCD	EN	35
LED	AZUL	12	Display_7_SEG.	D-2	25	Display_LCD	D4	30
LED_RGB	R	2	Display_7_SEG.	E-1	23	Display_LCD	D5	32
LED_RGB	G	4	Display_7_SEG.	F-9	26	Display_LCD	D6	34
LED_RGB	B	5	Display_7_SEG.	G-10	28	Display_LCD	D7	36
LED BRANCO	LOAD	13	Display_7_SEG.	PD-5	29			
DISTANCIA	TRIG	6	Componente		Pino			
DISTANCIA	ECHO	7	MOTOR DC		3			
TERMISTOR		A0	SERVO		1			
LDR		A1	BUZZER		8			
POTENCIÔMETRO		A5						
CHAVE TÁCTIL		14						

## **2.3 Criando um programa e enviando para a placa Arduino**

### **2.3.1 Configurações Iniciais do Ambiente.**

Estas configurações precisam ser feitas apenas na primeira vez que o ambiente for utilizado

Entre no menu <Configurações> e selecione a opção <Preferências>. Na Janela que abre, especifique:

#### **1. A localização do Compilador Arduino.**

- DB4K-SA** – Ao clicar em cima da linha para especificação do compilador, uma janela abrirá. Navegue até a pasta do Software Arduino (normalmente C:\Program Files (x86)\Arduino ou C:\Program Files\Arduino ) e selecione o arquivo **Arduino\_debug**.
- DB4K-CS** - Você deve digitar o caminho completo até o arquivo **arduino\_debug** (Windows) ou **arduino** (Linux), presente da pasta onde o Software Arduino foi instalado.

Exemplo:

- **Windows 64:** C:\Program Files (x86)\Arduino\arduino\_debug.exe
- **Windows 32:** C:\Program Files \Arduino\arduino\_debug.exe
- **Ubuntu:** /home/meu\_nome/Arduino-nightly/arduino (meu\_nome é o seu nome de usuário)

## 2. A Pasta do Projeto.

- a. **DB4K-SA** - Por padrão, a pasta do projeto é a pasta raiz do DB4K, ou seja: `C:\DB4K-SA`, mas você pode especificar outra pasta de sua preferência.
- b. **DB4K-CS** - Você deve digitar o caminho completo até a pasta DB4K-CS. Ex:  
**Windows:** `C:\DB4K-CS` | **Linux:** `/home/meu_nome/DB4K-CS`

### 2.3.2 Configurações da placa Arduino e da porta COM

Entre no menu <Configurações> e selecione a opção <Preferências>. Na Janela que abre, especifique:

1. A **Placa Arduino** desejada
2. A **Porta COM** a ser utilizada

Estas configurações têm que ser feitas sempre que você conectar uma placa Arduino no computador.

Obs. **Para se certificar de que o DB4K configurou a porta COM adequadamente**, clique sobre o nome da porta mesmo que ela já tenha sido definida automaticamente pelo DB4K.

### 2.3.3 Enviando um programa para a placa Arduino

Para construir um programa, escolha os blocos desejados no menu de blocos e os posicione na área de trabalho, encaixando-os na ordem desejada. Os blocos de estruturas condicionais usam os blocos de sensores como parâmetros condicionantes. Os códigos construídos com o DB4K rodam dentro de um loop infinito, ou seja, a sequência de comandos será executada indefinidamente (a não ser que se utilize o bloco “parar programa”, que é utilizado apenas em algumas atividades muito específicas).

Para carregar um programa na placa Arduino selecionada você deve montar o programa em blocos e apertar o botão laranja presente na parte superior da área de montagem de programas. Esse botão envia o programa diretamente para a placa Arduino configurada anteriormente.

Caso você deseje parar a execução do programa, aperte o botão vermelho presente na parte superior da área de montagem de programas. Este botão envia um “programa em branco” para a placa Arduino.

Enquanto o programa estiver sendo enviado, o botão de envio (ou parada), ficará cinza. É possível que, em algumas máquinas com Windows, o programa demore cerca de um minuto ou mais para ser carregado.

Existe uma área na parte inferior da janela do programa, chamada: Mensagens IDE Arduino, onde podem ser vistas as mensagens enviadas pela IDE.

**Vídeo explicativo de uso do ambiente** (<https://youtu.be/ZkclQNE0dEY>)

### 2.3.4 Possíveis Problemas

1. Em algumas situações, é necessário que a **configuração da Placa Arduino e da porta COM seja feita antes no Software Arduino** para que as configurações dessas informações, dentro do DB4K, tenham efeito.
2. Caso, por alguma razão, o **DB4K não consiga enviar o programa para a placa**, pode-se utilizar a opção de menu <Código Arduino> <Abrir código na IDE Arduino> para abrir o código construído no Software Arduino e então enviá-lo para a Placa Arduino.
3. Existem duas possibilidades de uso da **biblioteca Ultrassonic**. Caso ocorra um erro de chamada de função da biblioteca que você tem instalada, entre no arquivo `DB4K-SA\DB4K\db4k_setup.js` ou `DB4K-SA\DB4K\db4k_setup.js` e modifique a opção de Biblioteca. Lembre-se também de realizar as ações descritas no item 2.2.2.