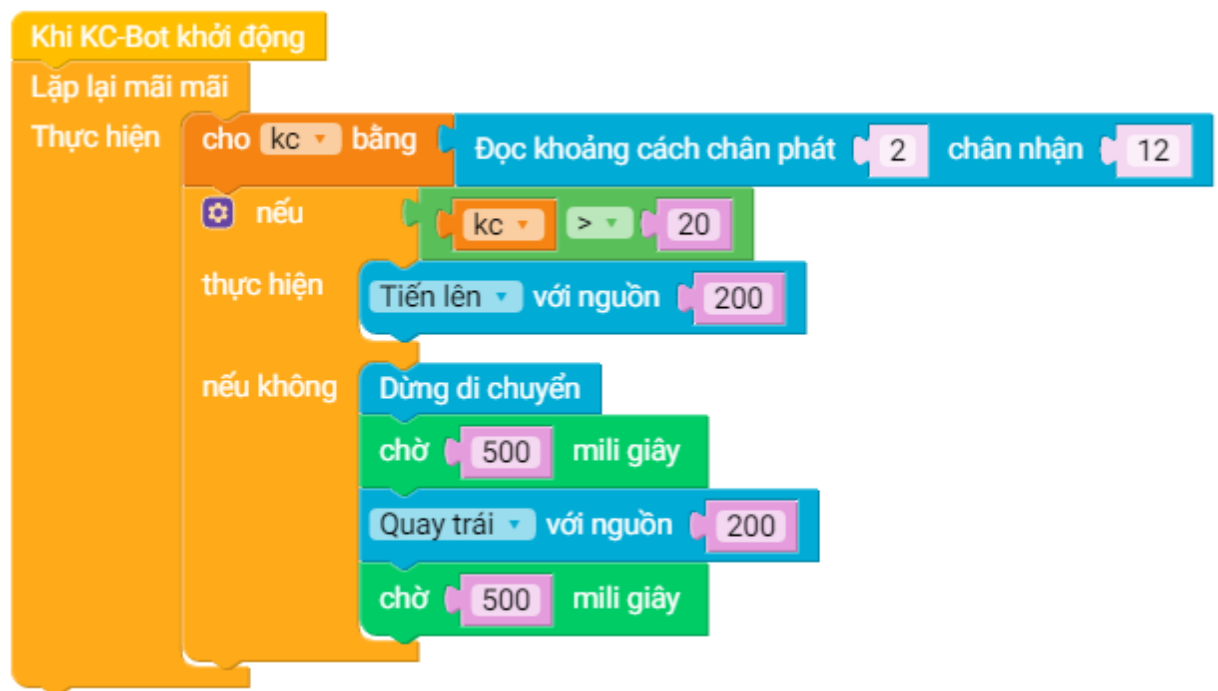


HƯỚNG DẪN SỬ DỤNG PHẦN MỀM EASYCODE



Mục Lục

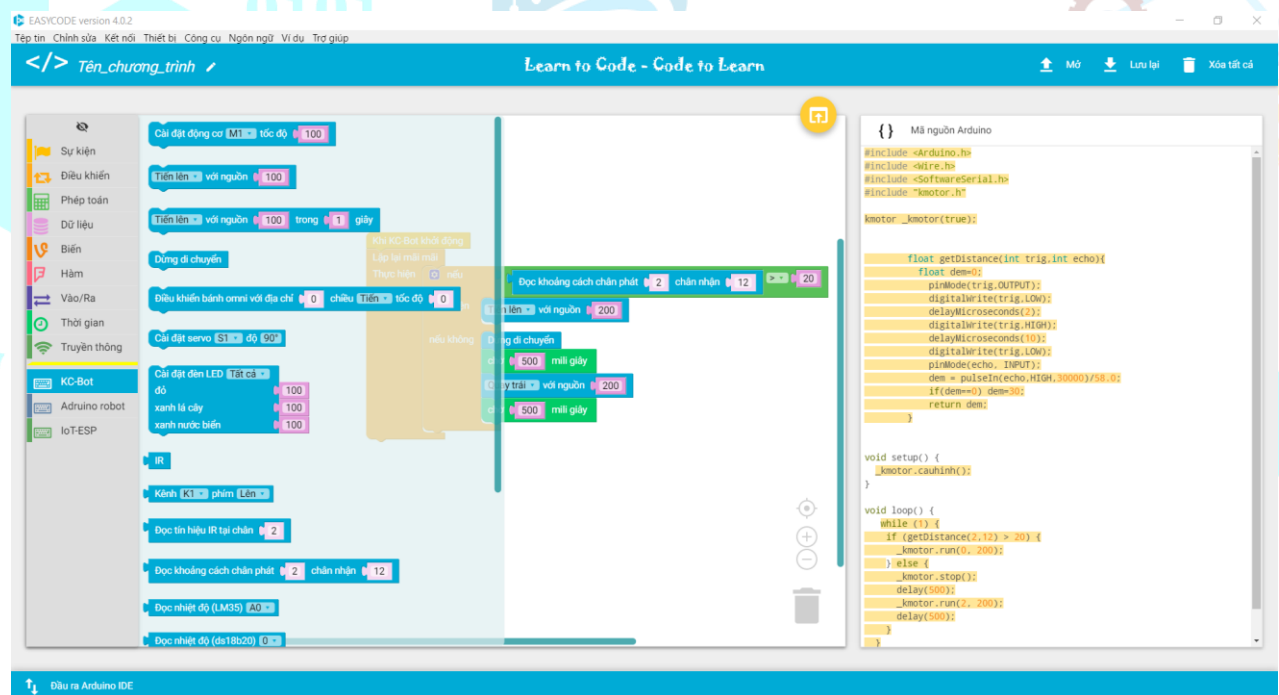
GIỚI THIỆU	3
HƯỚNG DẪN CÀI ĐẶT PHẦN MỀM	4
LÀM QUEN VỚI PHẦN MỀM EASYCODE	7
I. Giao diện làm việc	7
II. Nạp chương mẫu với phần mềm Easycode	7
III. Các câu lệnh trong phần mềm EasyCode	10
1. Khối lệnh: Sự kiện	10
2. Khối lệnh: Điều khiển.....	11
3. Khối lệnh: Phép toán	18
4. Khối lệnh: Dữ liệu	20
5. Khối lệnh: Biến.....	21
6. Khối lệnh: Hàm	22
7. Khối lệnh: Vào\ra.....	24
8. Khối lệnh: Thời gian.....	28
9. Khối lệnh: Truyền thông.....	29
10. Khối lệnh KC-Bot	31
11. Câu lệnh cho robot giáo dục Arduino robot.	33
12. Câu lệnh lập trình những hệ thống IOT.....	33
LỜI KẾT	34

GIỚI THIỆU

Phần mềm Easycode là phần mềm được CÔNG TY CỔ PHẦN PHÁT TRIỂN GIÁO DỤC KIDSCODE nghiên cứu và phát triển dựa trên mã nguồn mở của Ardublockly (<https://github.com/carlosperate/ardublockly>). Đây là phần mềm lập trình dành cho trẻ em và những người mới bắt đầu học lập trình, với ngôn ngữ kéo thả giúp các em dễ dàng tiếp cận với công nghệ và phát triển tư duy lập trình bằng ngôn ngữ lập trình cơ bản và đơn giản nhất.

Phần mềm được ứng dụng trong dạy STEM robot, giúp việc tiếp cận và lập trình điều khiển robot trở nên dễ dàng hơn. Nó còn được thiết kế để lập trình cho mạch Arduino và nhiều bo mạch phổ biến hiện nay.

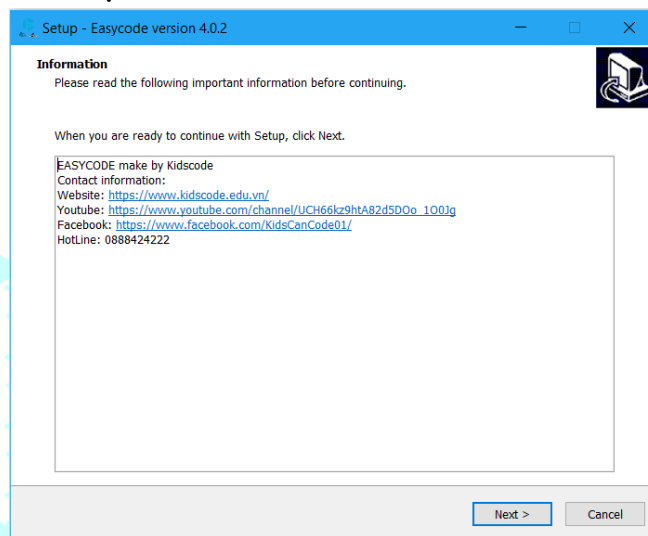
Dưới đây là giao diện chính của phần mềm Easycode khi khởi động:



Hình 1. Giao diện phần mềm Easycode

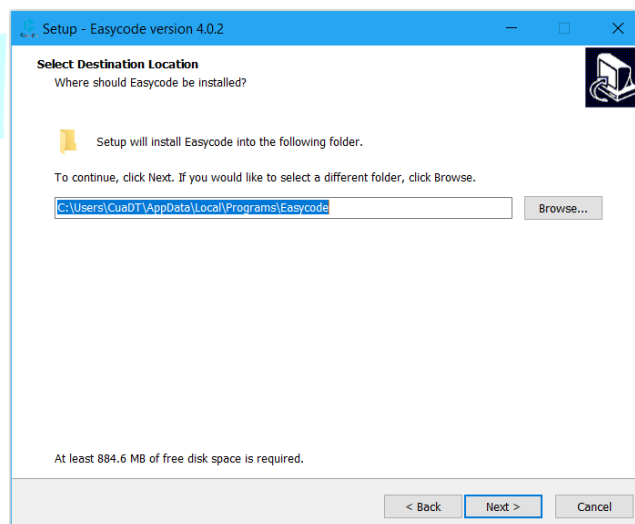
HƯỚNG DẪN CÀI ĐẶT PHẦN MỀM

1. Tải file cài đặt của Easycode trên trang: <https://www.kidscode.edu.vn/phan-mem>
(Hiện nay, Easycode mới chỉ hỗ trợ trên hệ điều hành Window, những hệ điều hành khác sẽ được cập nhật trong thời gian sắp tới)
2. Sau khi đã tải bản cài đặt các bạn nhấn đúp chuột trái để bắt đầu cài đặt (Các bạn lên tắt phần mềm diệt virus hiện có trên máy để đảm bảo quá trình cài đặt không bị lỗi).
3. Tại bảng thông tin các bạn chọn “Next”



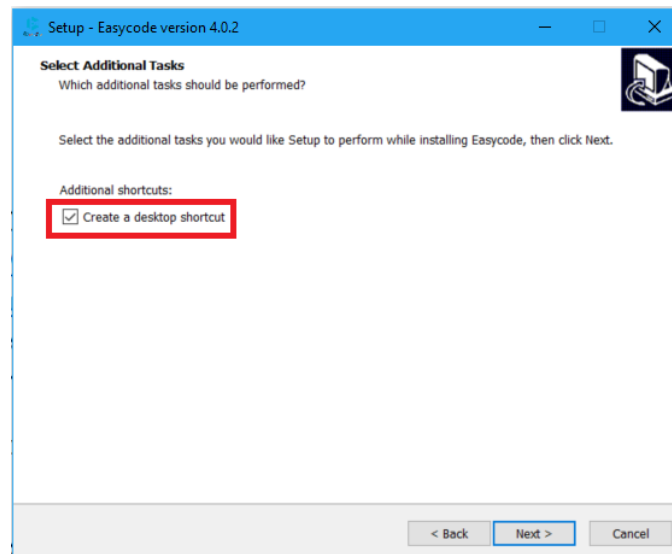
Hình 2. Bảng thông tin cài đặt

4. Tại khung cài đặt thứ 2 chọn “Next”



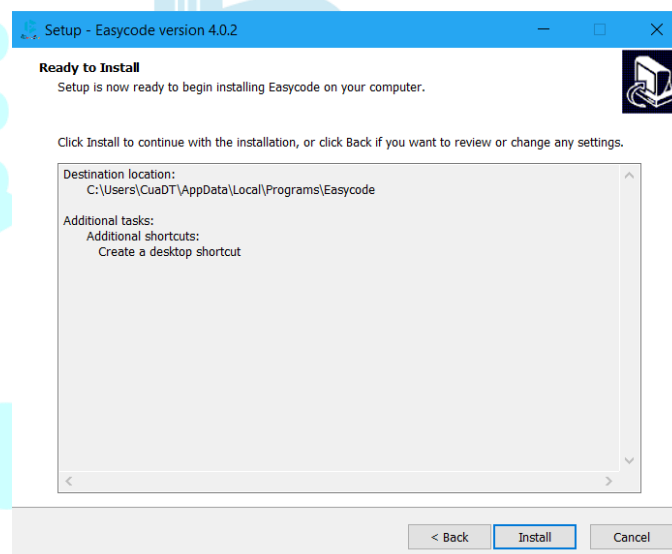
Hình 3. Chọn thư mục chứa phần mềm

5. Tại khung cài đặt tiếp theo các bạn chọn “Create a desktop shortcut” sau đó chọn “Next”



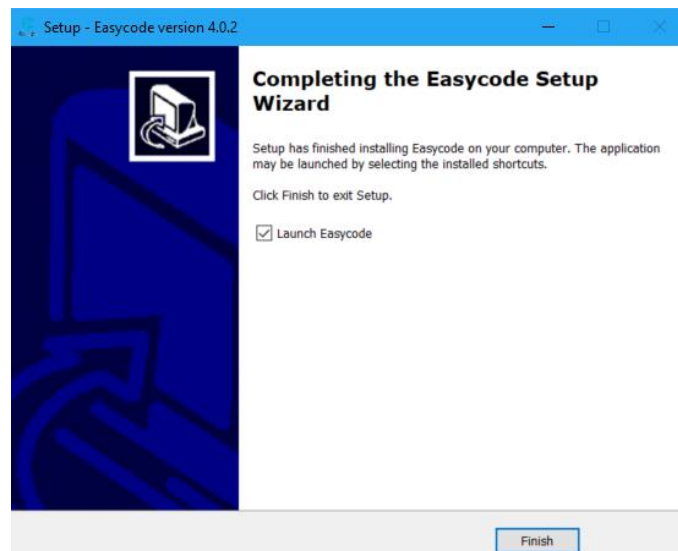
Hình 4. Tạo file shortcut

6. Tại khung cài đặt tiếp theo chọn “Install” để bắt đầu cài đặt



Hình 5. Bắt đầu cài đặt

7. Đợi phần mềm cài xong và chọn “Finish”



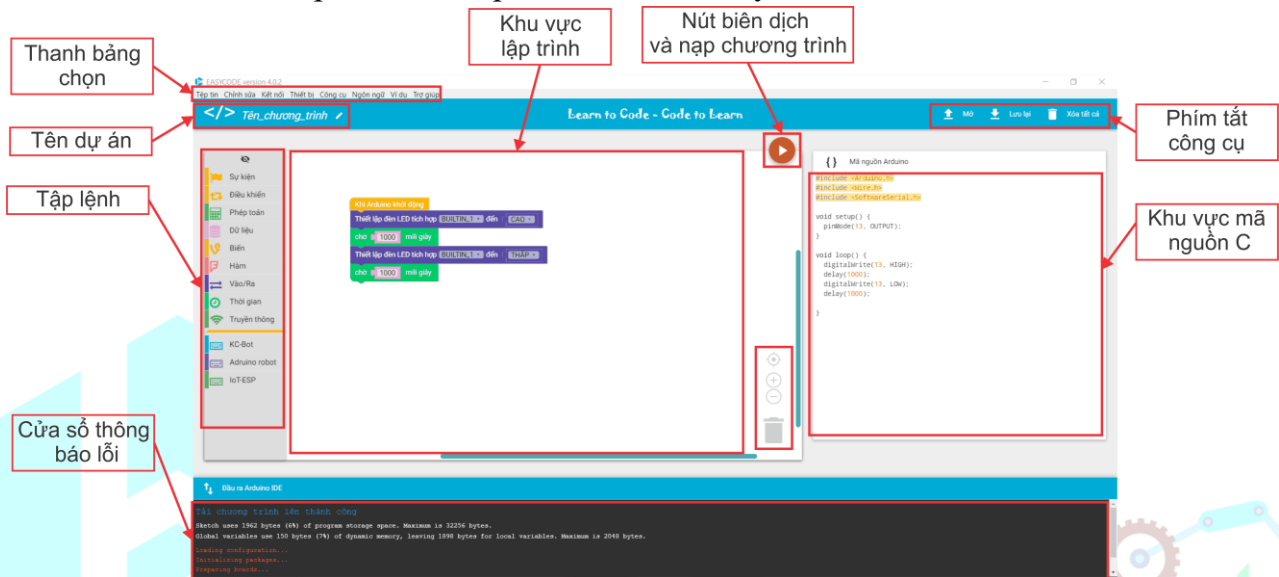
Hình 6. Hoàn thành cài đặt

Như thế chúng ta đã cài đặt xong phần mềm lập trình kéo thả Easycode.

LÀM QUEN VỚI PHẦN MỀM EASYCODE

I. Giao diện làm việc

Giao diện làm việc phần mềm lập trình kéo thả Easycode.



Hình 7. Giao diện làm việc phần mềm lập trình Easycode

Thanh bảng chọn: Gồm các tính năng như tệp tin, chỉnh sửa, kết nối, thiết bị, ngôn ngữ, ví dụ và trợ giúp về phần mềm.

Tên dự án: Tên cửa chương trình đang mở.

Tập lệnh: Nơi chứa các khối lệnh để lập trình. Các khối được chia thành các tập hợp theo chức năng.

Cửa sổ thông báo lỗi: Hiện thị thông tin về quá trình nạp và thông tin về việc kiểm tra chương trình

Khu vực lập trình: Lập trình, kéo thả và ghép các khối lệnh để tạo ra một chương trình.

Nút biên dịch và nạp chương trình: Dùng để kiểm tra lỗi chương trình hoặc nạp chương trình xuống mạch điều khiển.

Phím tắt công cụ: Bao gồm 3 công cụ thường dùng như: “Mở” (mở một chương trình mới); “Lưu lại” (Lưu lại chương trình hiện tại); “Xóa tất cả” (Xóa tất cả các lệnh có trên vùng lập trình).

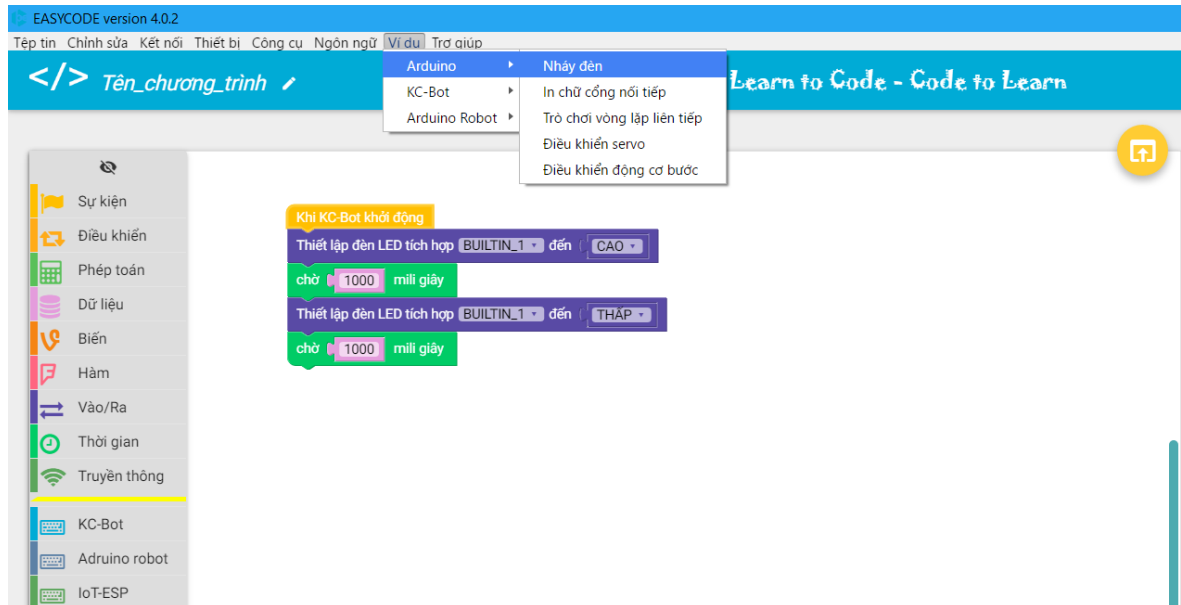
Khu vực mã nguồn C: Hiện thị chương trình bằng mã nguồn C được sinh ra tương ứng với từng khối lệnh trong khu vực lập trình.

II. Nạp chương mẫu với phần mềm Easycode

Phần mềm Easycode hiện có thể sử dụng để lập trình cho mạch STEM board được phát triển bởi Công ty cổ phần Phát triển Giáo dục Kidscodex và những bo mạch trên nền tảng Arduino phổ biến như: STEM board, Arduino nano, Arduino uno, Arduino mega.

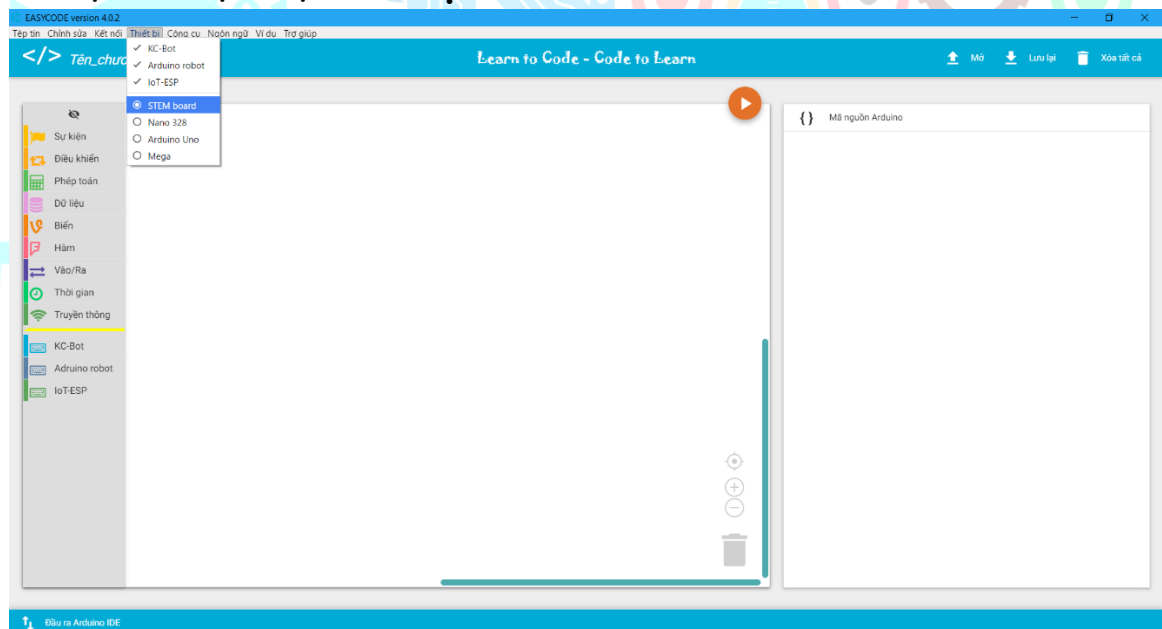
Để nạp 1 chương trình vào mạch điều khiển chúng ta thao tác qua các bước:

Bước 1: Tạo một chương trình, ở đây chúng ta sử dụng một chương trình mẫu. Ví dụ chương trình có tên “nhảy đèn”: Chọn **Ví dụ** → **Arduino** → **Nhảy đèn**.



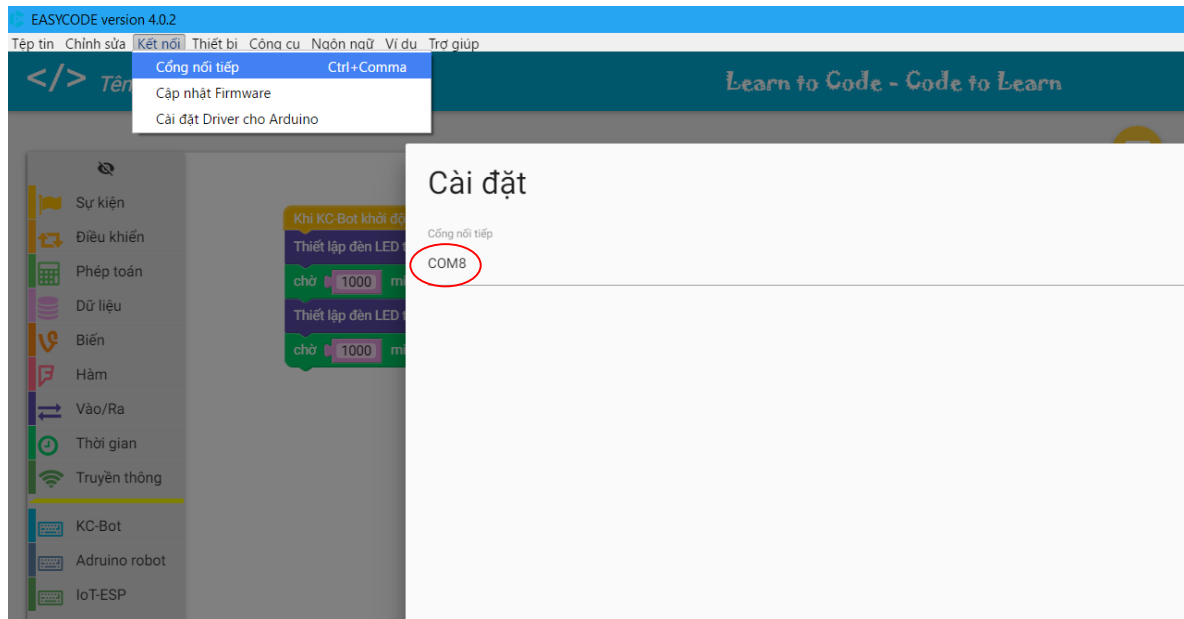
Hình 8: Chọn ví dụ

Bước 2: Chọn thiết bị : Chọn **Thiết bị** → **STEM board**



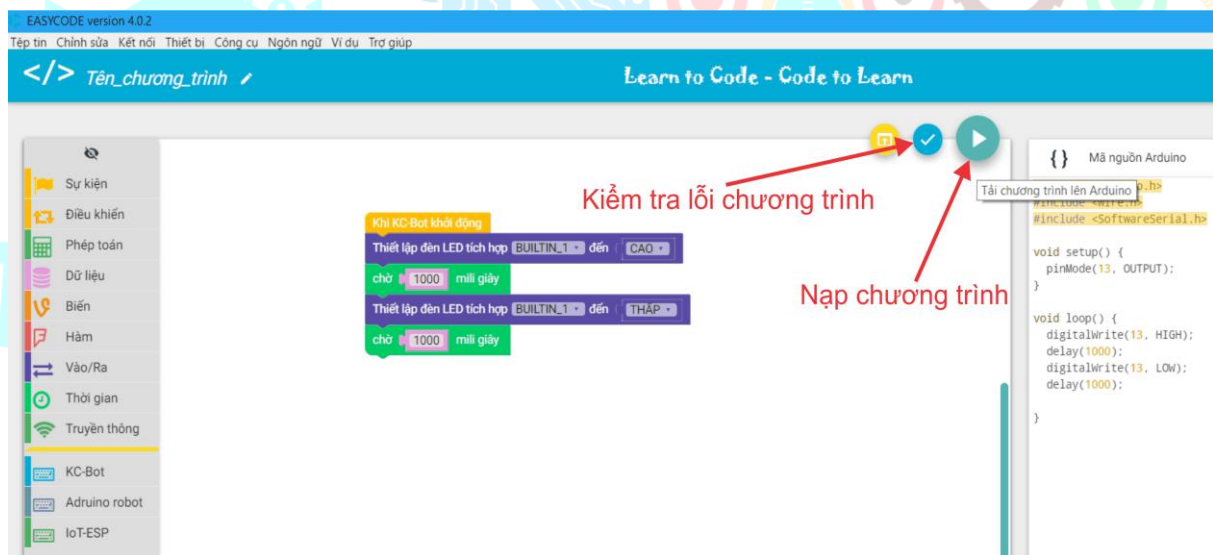
Hình 9: Chọn thiết bị

Bước 3: Chọn cổng COM kết nối đến mạch điều khiển: Chọn **Kết nối** → **Cổng kết nối**. Sau đó nhấn vào cổng COM của mạch điều khiển.



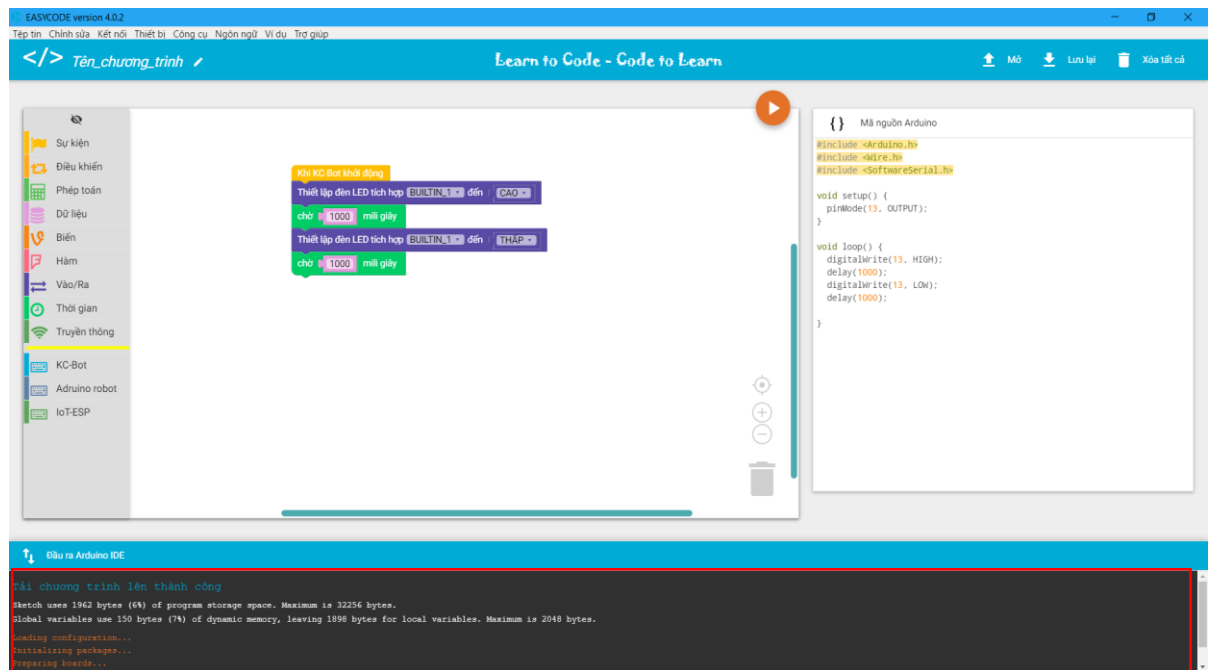
Hình 10: Chọn cổng COM kết nối

Bước 4: Chọn nút  để nạp chương trình.



Hình 11: Nút biên dịch và nạp chương trình

Bước 5: Kiểm tra thông báo trong phần cửa sổ thông báo lỗi.



Hình 12: Nạp chương trình thành công

Với ví dụ “nháy đèn”, khi nạp xong chương trình đèn Led được kết nối sẵn trên mạch sẽ nháy với tần số 1 giây một lần.

Lưu ý: khi chúng ta kéo một câu lệnh ra khung làm việc mặc định câu lệnh sẽ được đặt vào vòng lặp mãi mãi của chương trình. Để viết một chương trình chạy một lần duy nhất. (Xem mục 6.1)

III. Các câu lệnh trong phần mềm EasyCode

1. Khối lệnh: Sự kiện

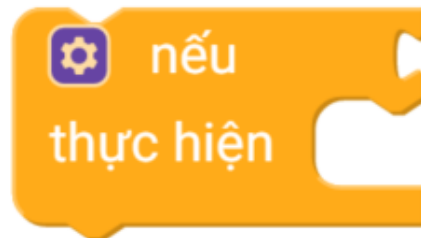
Khối lệnh “Sự kiện” chứa các câu lệnh để bắt đầu chương trình hoặc xác định sự kiện bắt đầu một chuỗi câu lệnh lập trình.

Khối lệnh	Ý nghĩa
Khi KC-Bot khởi động	Câu lệnh bắt đầu chương trình với KC-Bot
Khi Arduino robot khởi động	Câu lệnh bắt đầu chương trình với Arduino robot
Khi Arduino khởi động	Câu lệnh bắt đầu chương trình với Arduino

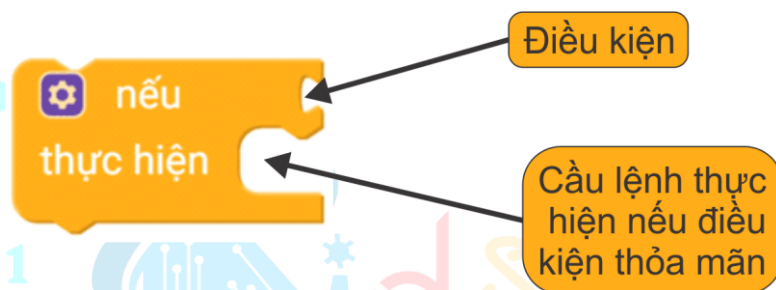
2. Khối lệnh: Điều khiển

Khối lệnh chứa các câu lệnh điều kiện và rẽ nhánh.

2.1. Câu lệnh: nếu-thực hiện



Ý nghĩa: Câu lệnh này dùng để kiểm tra một điều kiện nào đó nếu thỏa mãn điều kiện thì thực hiện câu lệnh.



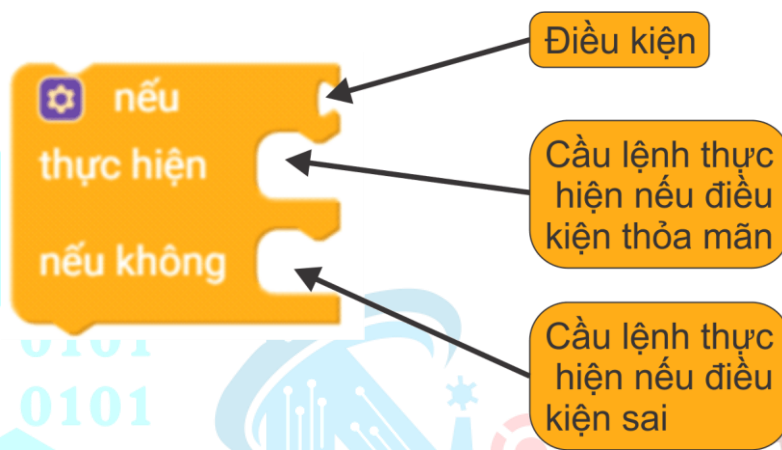
Ví dụ 2.1: Chúng ta sẽ sử dụng hai linh kiện là nút bấm và đèn LED, nút bấm được nối đến chân số 2 trên mạch điều khiển của chúng ta. Ở đây chúng ta kết nối mạch điện để khi bấm nút thì chân số 2 nhận mức cao và khi không bấm nút chân số 2 nhận mức thấp, đèn LED đã được tích hợp sẵn trên mạch và được kết nối đến chân 13, nếu chân 13 có mức cao đèn sáng mức thấp thì đèn tắt. Chương trình sẽ hoạt động nếu như bấm nút thì đèn sáng, không bấm thì đèn tắt.



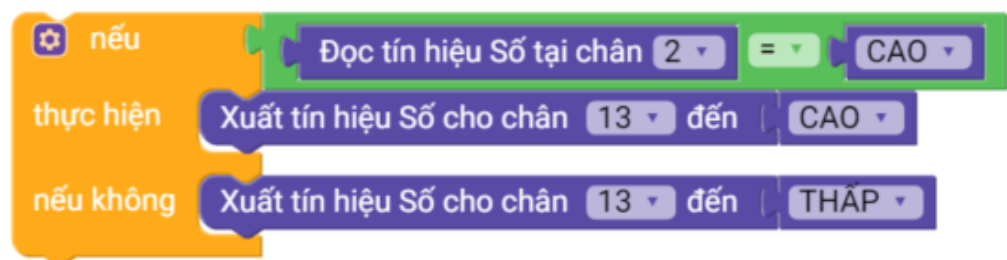
2.2. Câu lệnh: nếu - thực hiện – nếu không



Ý nghĩa: Câu lệnh này dùng để kiểm tra một điều kiện nào đó, nếu điều kiện đó thỏa mãn thì thực hiện câu lệnh bên trên, nếu không thỏa mãn thì thực hiện câu lệnh ở bên dưới.



Ví dụ 2.2: Yêu cầu giống ví dụ 2.1

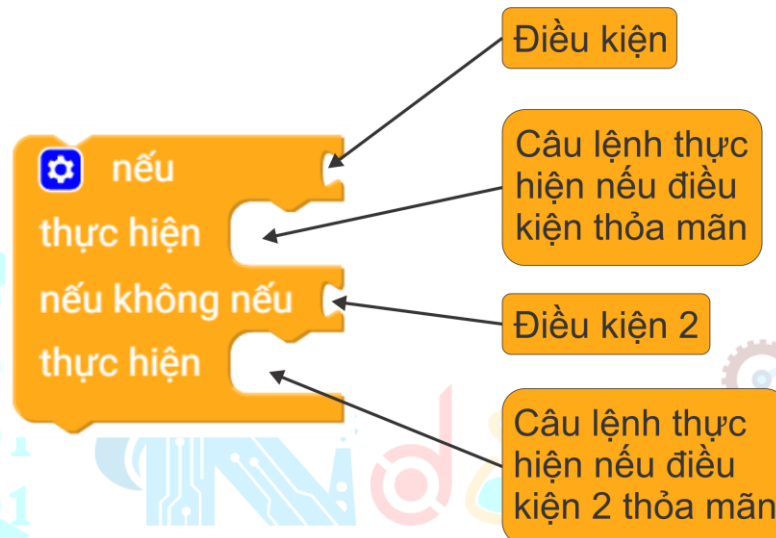


Ở đây chúng ta đã không cần sử dụng câu lệnh điều kiện thứ 2 do câu lệnh “nếu không” đã là câu lệnh phủ định của câu lệnh “nếu” phía trên.

2.3. Câu lệnh: nếu – thực hiện, nếu không nếu – thực hiện.



Ý nghĩa: Câu lệnh này dùng để kiểm tra một điều kiện nào đó, nếu thỏa mãn điều kiện thì thực hiện câu lệnh bên trên. Nếu không thỏa mãn sẽ kiểm tra điều kiện thứ 2, thỏa mãn điều kiện đó thì thực hiện câu lệnh ở về bên dưới.



Ví dụ 2.3: Từ ví dụ 2.1 kết nối thêm một nút bấm tại chân số 3 của mạch điều khiển sao cho, khi bấm nút thì chân số 3 nhận mức cao và không bấm nút chân số 3 nhận mức thấp. Chương trình thực hiện để khi bấm nút tại chân số 2 thì đèn tại chân số 13 sáng, nếu bấm nút tại chân số 3 thì đèn tại chân số 13 tắt.



2.4. Câu lệnh: Lặp lại mãi mãi

Lặp lại mãi mãi
Thực hiện

Ý nghĩa: Câu lệnh này để lặp lại chương trình nào đó với số lần thực hiện là mãi mãi.

Lặp lại mãi mãi
Thực hiện

Câu lệnh cần
thực hiện

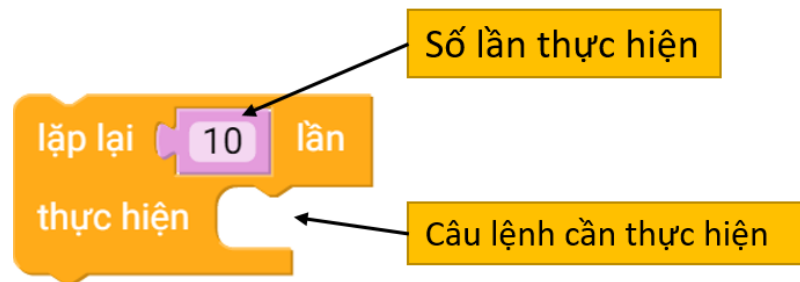
Ví dụ 2.4: Chương trình thực hiện để bật LED tại chân 13 sáng 1s sau đó tắt 1s và lặp lại quá trình này mãi mãi.



2.5. Câu lệnh: Lặp lại với N lần. (N chạy từ là số nguyên dương)

lặp lại 10 lần
thực hiện

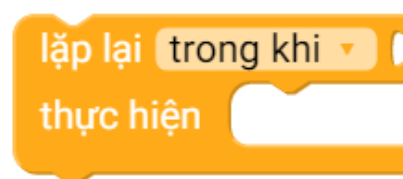
Ý nghĩa: Câu lệnh này để thực hiện một chương trình nào đó với số lần thực hiện N lần



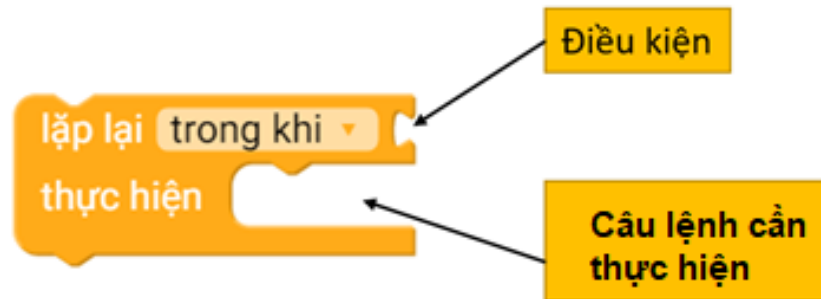
Ví dụ 2.5: Chương trình điều khiển đèn LED tại chân 13 sáng 1s sau đó tắt 1s lặp lại 5 lần sau đó điều khiển đèn LED sáng 100 milis sau đó tắt 100 milis và lặp lại 5 lần.



2.6. Câu lệnh: lặp lại – trong khi



Ý nghĩa: Câu lệnh sử dụng để lặp lại những câu lệnh bên trong nếu điều kiện đúng. Điều kiện sai sẽ thoát khỏi vòng lặp.



Ví dụ 2.6: Kết nối như ví dụ 2.1. Ở đây chúng ta lập trình điều khiển, khi không bấm nút thì đèn tắt, bấm nút đèn sẽ sáng đến khi bỏ tay ra.



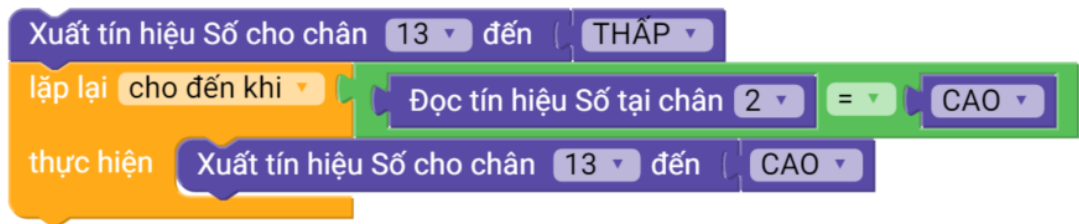
Ở đây chúng ta hiểu là khi chúng ta không bấm nút thì chương trình sẽ thực hiện câu lệnh xuất ra chân 13 mức thấp lên tục. Khi chúng ta bấm nút chương trình sẽ chỉ thực hiện câu lệnh bên dưới là xuất ra chân 13 mức cao mà không thực hiện câu lệnh xuất ra chân 13 mức thấp.

2.7. Câu lệnh: Lặp lại – cho đến khi



Ý nghĩa: Câu lệnh sử dụng để lặp lại những câu lệnh bên trong nếu điều kiện sai. Điều kiện đúng sẽ thoát khỏi vòng lặp.

Ví dụ 2.7: Kết nối giống ví dụ 2.1. Chương trình điều khiển khi không bấm nút thì đèn sáng, khi bấm nút thì đèn tắt.

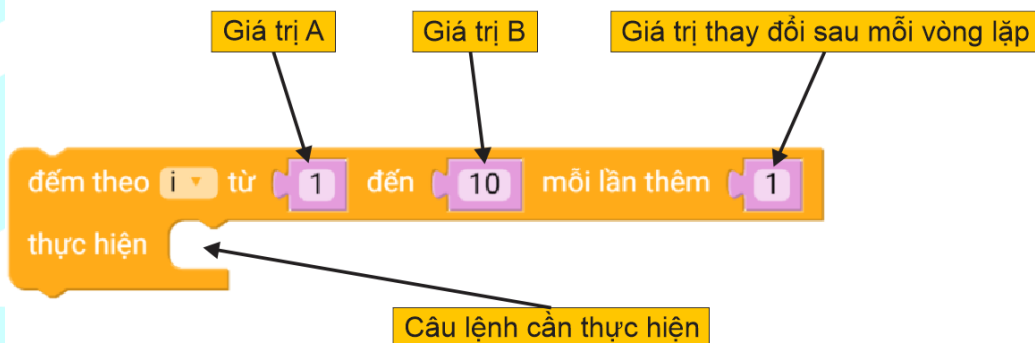


Ở đây chúng ta thấy khi chúng ta không bấm nút (điều kiện chân số 2 sai) thì một câu lệnh bên trong khối lệnh điều kiện sẽ được thực hiện, còn khi ta bấm nút (điều kiện đúng) thì câu lệnh bên ngoài sẽ được thực hiện.

2.8. Câu lệnh: Đếm



Ý nghĩa: Câu lệnh lặp lại số lần theo giá trị thay đổi từ A đến B theo giá trị thay đổi sau mỗi vòng lặp, ban đầu giá trị “i” được khởi tạo bằng A vòng lặp dừng lại khi giá trị i lớn hơn hoặc bằng giá trị B.



Ví dụ 2.8: Thực hiện lệnh nhấp nháy đèn LED sáng 1s tắt 1s với “i” chạy từ giá trị 0 đến 10 và giá trị thay đổi là 2. Sau đó thực hiện nhấp nháy đèn LED sáng 100 milis tắt 100 milis với i chạy từ 0 đến 10 đơn vị tăng là 1.



Ở đây chúng ta sẽ thấy LED nhấp nháy với tần số 1s sẽ chỉ lặp lại 5 lần. Còn với tần số 100 milis số lần lặp lại sẽ là 10 lần.

3. Khối lệnh: Phép toán

3.1. Câu lệnh: Toán tử số học



Ý nghĩa: Câu lệnh này sẽ thực hiện phép toán tổng của hai số, câu lệnh chứa các phép toán cộng, trừ, nhân, chia, mũ.



Ví dụ 3.1:

cho item ▾ bằng 5 + 7

cho item ▾ bằng item ▾ - 7

cho item ▾ bằng item ▾ ÷ item ▾

3.2. Câu lệnh: Toán tử so sánh



Ý nghĩa: Câu lệnh sử dụng các toán tử so sánh hỗ trợ các câu lệnh điều kiện.



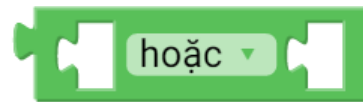
Ví dụ: Sử dụng câu lệnh để kiểm tra tín hiệu tại chân số 2. Kết nối phân cứng theo ví dụ 2.1

```

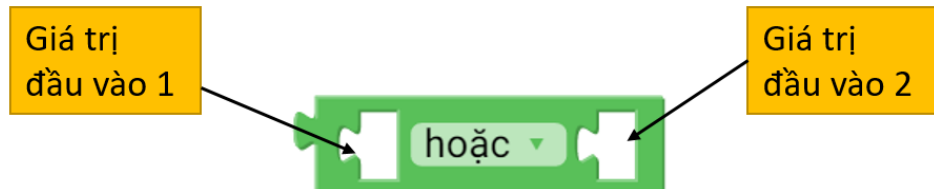
nếu [Độc tín hiệu Số tại chân 2 ▾] = CAO ▾
thực hiện [Xuất tín hiệu Số cho chân 13 ▾] đến CAO ▾

nếu [Độc tín hiệu Số tại chân 2 ▾] = THẤP ▾
thực hiện [Xuất tín hiệu Số cho chân 13 ▾] đến THẤP ▾
  
```

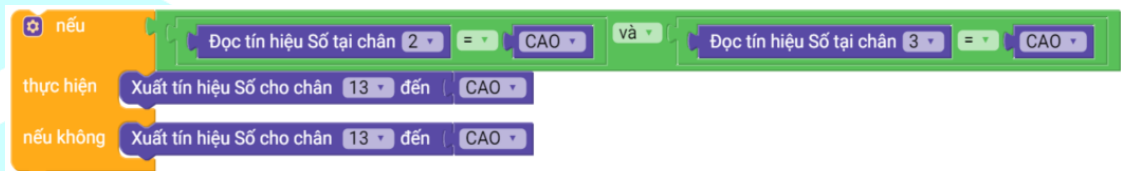
3.3. Câu lệnh: Toán tử Logic



Ý nghĩa: Sử dụng để kết hợp các điều kiện đồng thời:



Ví dụ 3.3: Kết nối theo ví dụ 2.3. chương trình thực hiện khi bấm cả hai nút tại chân 2 và chân 3 thì đèn mới sáng.

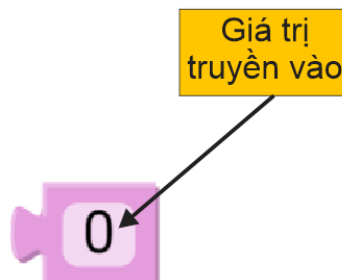


4. Khối lệnh: Dữ liệu

4.1. Câu lệnh: Số học



Ý nghĩa: Câu lệnh tạo một giá trị số học.



Ví dụ 4.1:



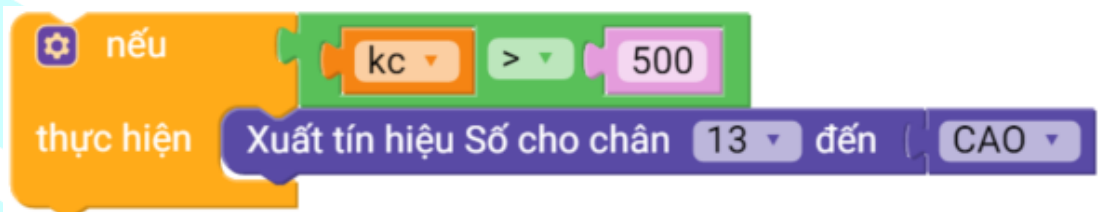
5. Khối lệnh: Biến

5.1. Câu lệnh: Lấy biến

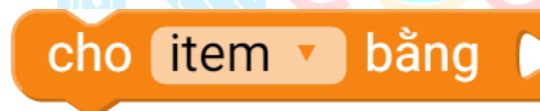


Ý nghĩa: Câu lệnh lấy ra biến đã được tạo để sử dụng

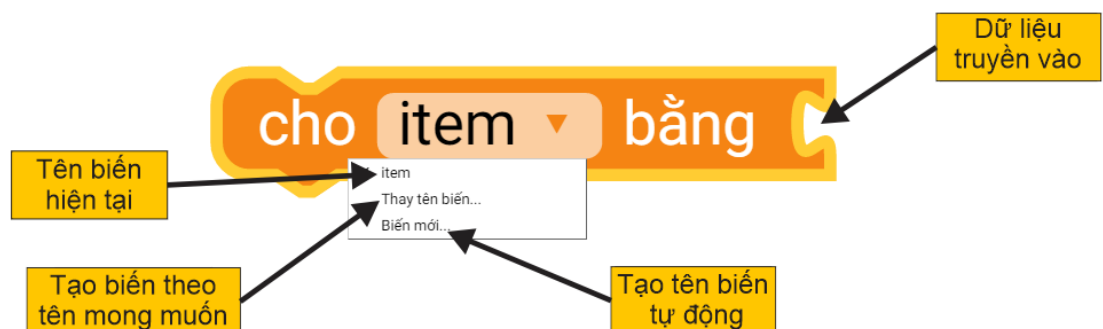
Ví dụ 5.1: Chúng ta kiểm tra giá trị của biến “kc” nếu > 500 thì xuất tín hiệu ra chân số 13 mức cao.



5.2. Câu lệnh: Tạo biến



Ý nghĩa: Tạo một biến mới để sử dụng theo ý người dùng. Lưu ý khối này chúng ta sẽ không sử dụng được lệnh copy/paste nếu muốn tạo biến mới chúng ta cần lấy lại khối lệnh trong tệp lệnh.

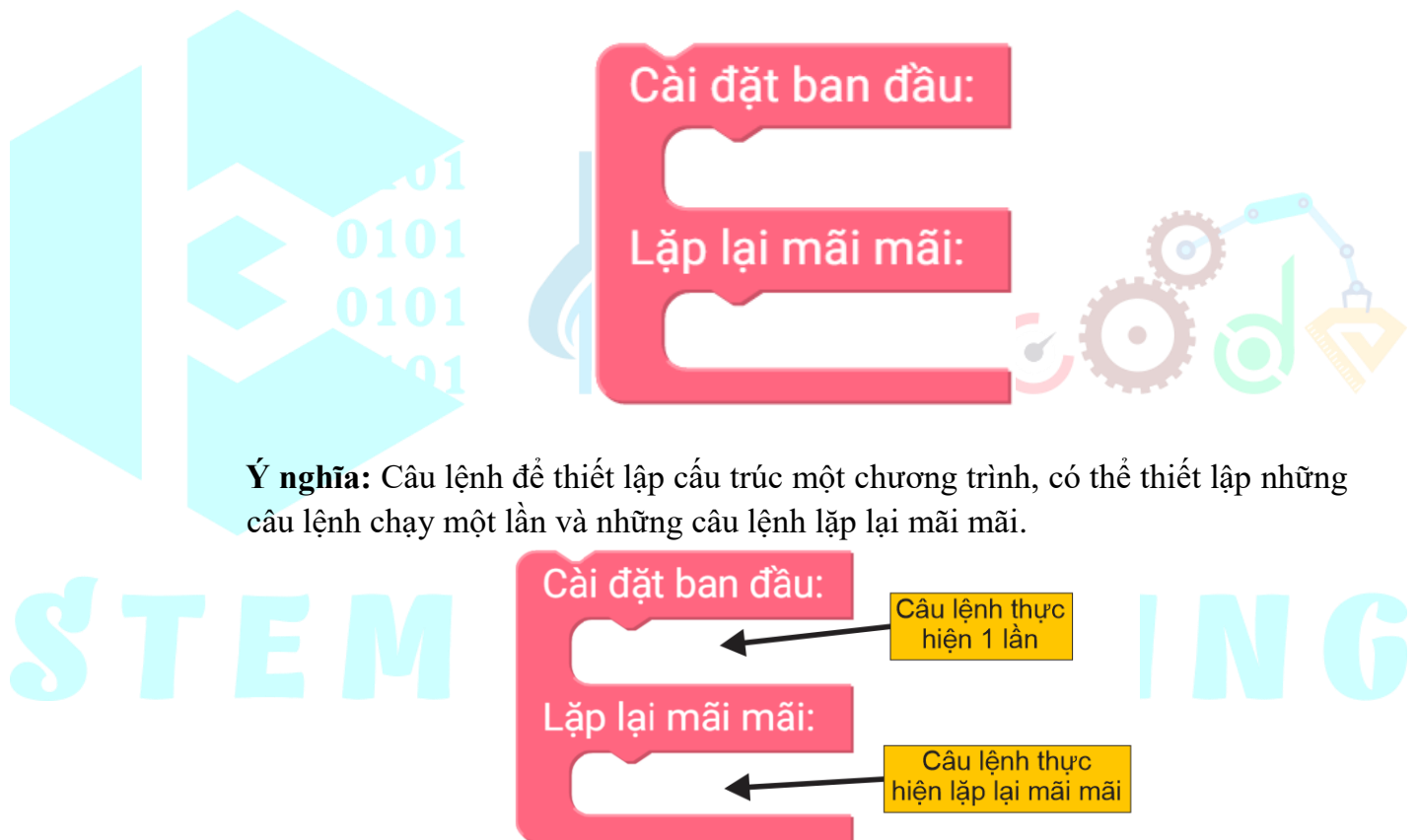


Ví dụ 5.2: Tạo một biến mới với dữ liệu truyền vào là dạng số học



6. Khối lệnh: Hàm

6.1. Câu lệnh: Cấu trúc chương trình.



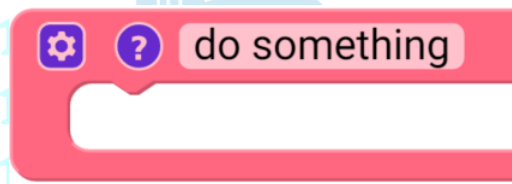
Ý nghĩa: Câu lệnh để thiết lập cấu trúc một chương trình, có thể thiết lập những câu lệnh chạy một lần và những câu lệnh lặp lại mãi mãi.

Ví dụ 6.1: Lập trình để đèn LED 13 nhấp nháy một lần sau đó sáng mãi.

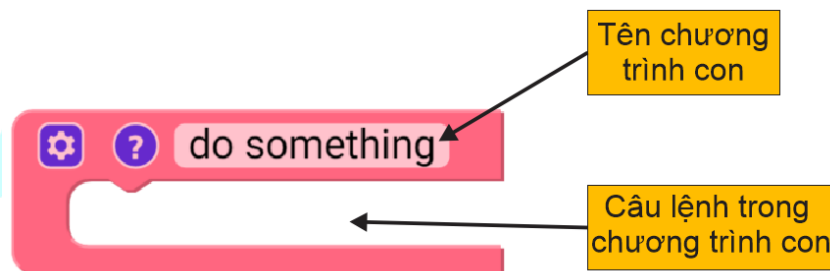


Trong chương trình này chúng ta thấy với về chương trình phía trên sẽ được chạy một lần khi khởi động mạch điều khiển sau đó chương trình sẽ lặp lại về thứ 2 mãi mãi.

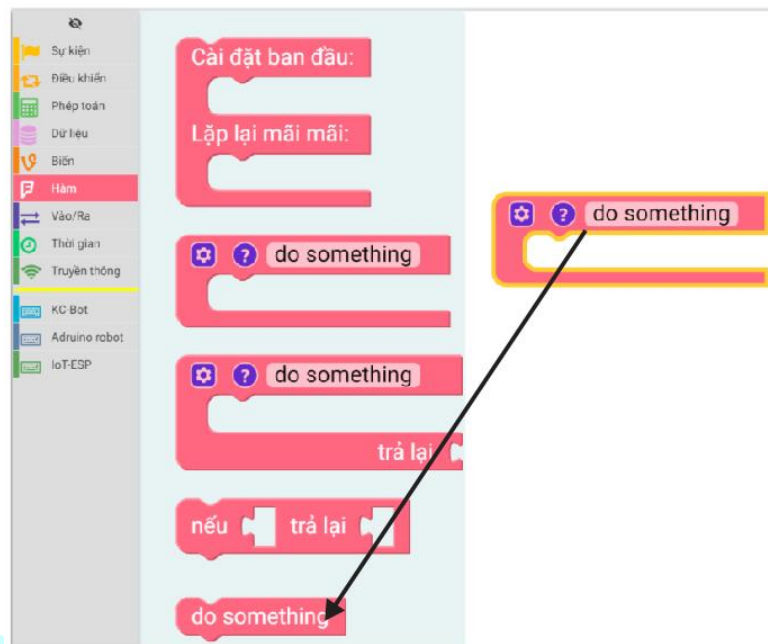
6.2. Tạo hàm con



Ý nghĩa: Tạo một hàm con để có thể sử dụng trong chương trình lập trình.



Khi lệnh tạo khối được kéo ra, một khối con sẽ được tạo trong tệp lệnh “hàm”.



Ví dụ 6.3: Tạo một chương trình con điều khiển đèn nhấp nháy.

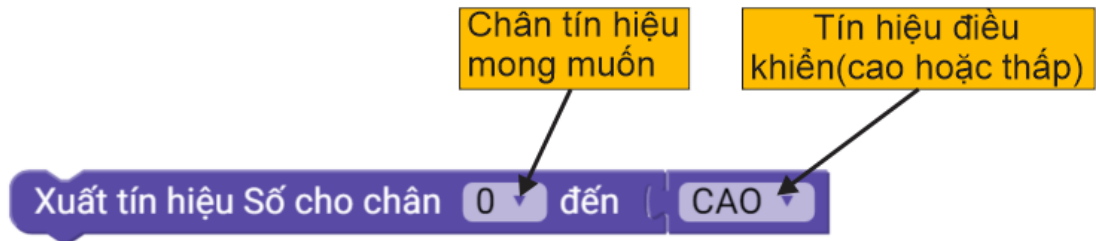


7. Khối lệnh: Vào\ra

7.1. Câu lệnh: Xuất tín hiệu Số



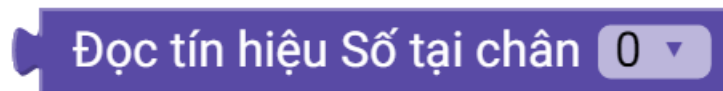
Ý nghĩa: Xuất giá trị cao hoặc thấp (tương ứng 0V đến 5V) cho chân được chọn. Có 14 chân digital từ cổng 0 đến cổng 13, và 6 cổng analog từ cổng A0 đến A5.



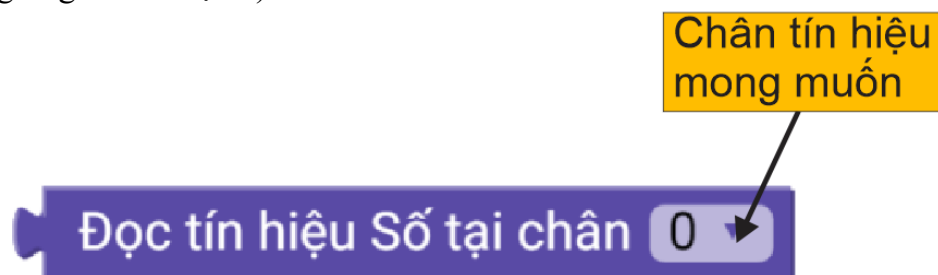
Ví dụ 7.1: Lập trình điều khiển đèn LED tại chân 13 nhấp nháy theo tần số sáng 1s, tắt 1s.



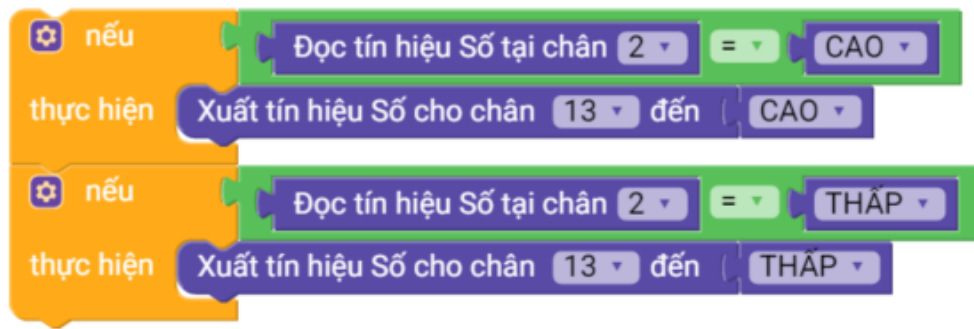
7.2. Câu lệnh: Đọc tín hiệu số.



Ý nghĩa: Đọc tín hiệu tại chân mong muốn. Tín hiệu đọc về sẽ là cao hoặc thấp (tương ứng với 0 hoặc 1)



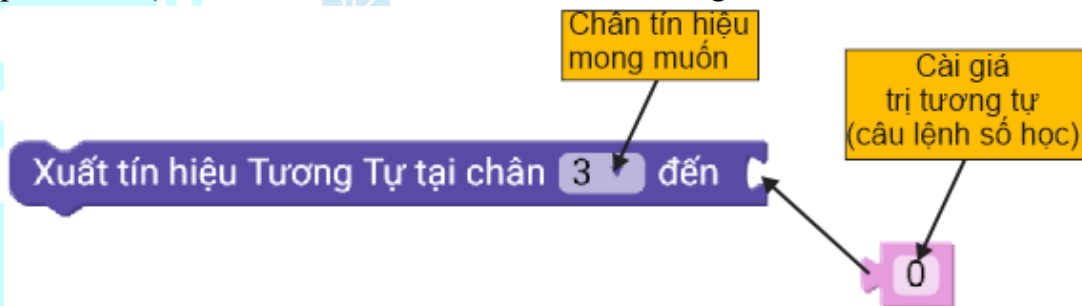
Ví dụ 7.2: Kết nối theo ví dụ 2.1. Ở đây chúng ta sẽ đọc tín hiệu tại chân số 2 nếu bấm nút thì đèn sáng, không bấm thì đèn tắt.



7.3. Câu lệnh: Xuất tín hiệu tương tự.

Xuất tín hiệu Tương Tự tại chân 5 đến

Ý nghĩa: xuất giá trị tương tự trong khoảng 0 đến 255 (tương ứng với mức điện áp 0 đến 5V) vào một chân PWM. Có 6 chân PWM gồm: 3, 5, 6, 9, 10, 11.

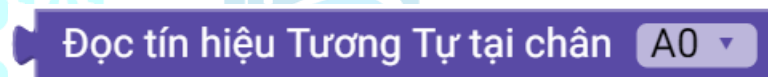


Ví dụ 7.3: Trên mạch điều khiển kết nối đèn LED đến chân 11 sao cho cực dương nối vào chân 11 và chân GND của đèn LED nối đến chân GND của mạch điều khiển. Lập trình điều khiển độ sáng đèn led tăng dần và giảm dần.



Ở đây chúng ta sử dụng 2 câu lệnh đếm để tạo ra vòng lặp đếm từ 0 đến 255 với giá trị tăng là 3 và giảm từ 255 về 0 với giá trị thay đổi mỗi lần là -3 sau đó chúng ta xuất chính giá trị đó ra tại chân số 11 để điều khiển đèn led.

7.4. Câu lệnh: Đọc tín hiệu tương tự



Ý nghĩa: Câu lệnh đọc về giá trị tương tự tại chân mong muốn, giá trị đọc về sẽ chạy từ 0 đến 1024 tương ứng với 0V đến 5V. Các chân có thể đọc tín hiệu tương tự là từ A0 đến A7.

Chân tín hiệu



Ví dụ 7.4: Kết nối biến trở (linh kiện sử dụng để thay đổi điện trở) đến chân A2 và đèn LED đến vị chân 11 theo ví dụ 7.3. Lập trình để sử dụng biến trở điều khiển độ sáng đèn.



Ở đây, đầu tiên chúng ta đọc tín hiệu tương tự tại chân A2 và ghi vào biến BT, sau đó chúng ta sử dụng công thức toán học để đổi từ thang 1023 đầu vào sang thang 255 đầu ra (xem phần 7.3) và ghi vào biến DS. Sau đó ghi giá trị biến DS ra chân 11.

7.5. Ví dụ: Tín hiệu số.



Ý nghĩa: Mức tín hiệu số của chân tín hiệu có thể dùng trong câu lệnh so sánh hoặc điều khiển tín hiệu ra chân có 2 mức tín hiệu là cao và thấp.



Ví dụ 7.5: Kết nối theo ví dụ 2.1



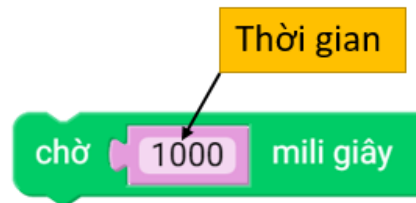
Ở đây chúng ta sẽ sử dụng câu lệnh mức để so sánh với tín hiệu nhận được tại chân số 2.

8. Khối lệnh: Thời gian

8.1. Câu lệnh: Chờ - mili giây



Ý nghĩa: Thời gian chờ thực hiện giữa các câu lệnh trong chương trình cụ thể tính bằng mili giây (10^{-3} giây)



Ví dụ 8.1: Điều khiển đèn LED tại chân 13 nhấp nháy theo tần số sáng 2s tắt 500mili

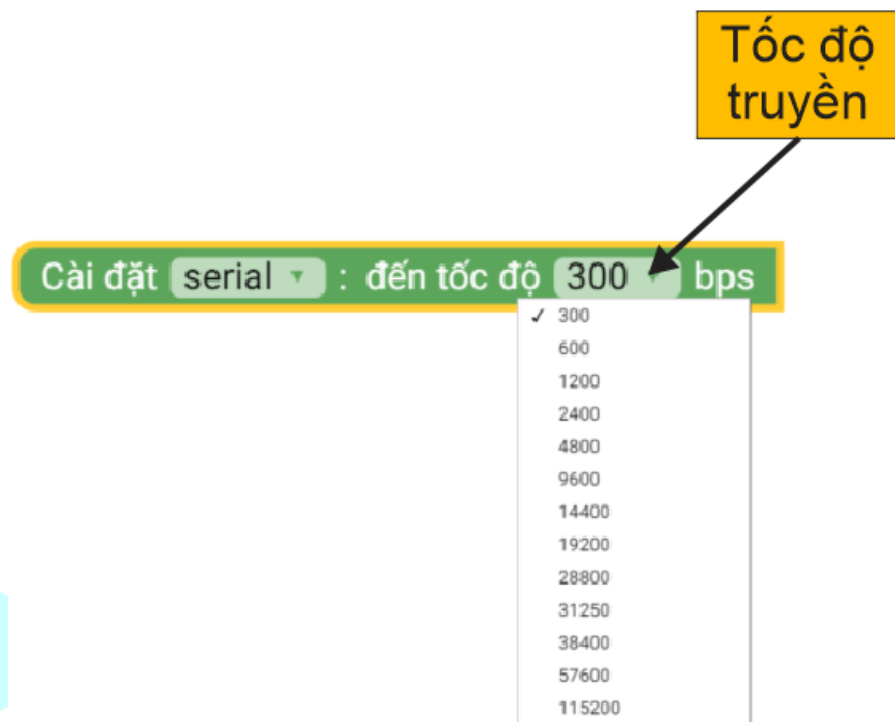


9. Khối lệnh: Truyền thông

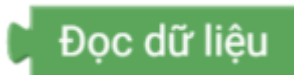
9.1. Câu lệnh: Cài đặt tốc độ truyền dữ liệu



Ý nghĩa: Câu lệnh sử dụng để cài đặt tốc độ truyền dữ liệu giữa các thiết bị với nhau như bluetooth hay wifi hay kết nối trực tiếp đến máy tính.



9.2. Câu lệnh: Đọc dữ liệu



Ý nghĩa: Câu lệnh giúp nhận dữ liệu gửi về từ các thiết bị truyền thông như Wifi, bluetooth hay máy tính theo chuẩn serial.

Ví dụ 9.2: Đọc dữ liệu và ghi vào biến.

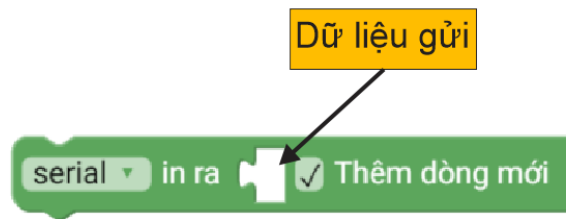


9.3. Câu lệnh: gửi dữ liệu

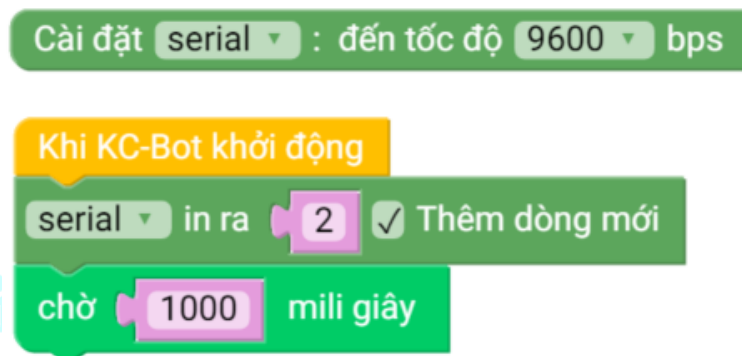


Ý nghĩa: Câu lệnh “gửi dữ liệu” giúp gửi các dữ liệu lên các thiết bị sử dụng chuẩn giao tiếp serial như Wifi, bluetooth hay máy tính. Dữ liệu gửi đi theo nhiều dạng số, ký tự . . .

Lưu ý: Chúng ta cần sử dụng câu lệnh “cài đặt tốc độ truyền dữ liệu” (mục 9.1) nếu không câu lệnh sẽ báo lỗi.



Ví dụ 9.3: Chúng ta sẽ gửi dữ liệu số ‘2’ từ mạch điều khiển lên máy tính với tần số 1s 1 lần.



Để kiểm tra dữ liệu gửi lên chúng ta sử dụng công cụ biểu đồ trên phần mềm. Để mở “Biểu đồ” chúng ta vào **Công cụ → Biểu đồ** (hoặc nhấn phím tắt shift t) giao diện sẽ được mở ra.



10. Khối lệnh KC-Bot

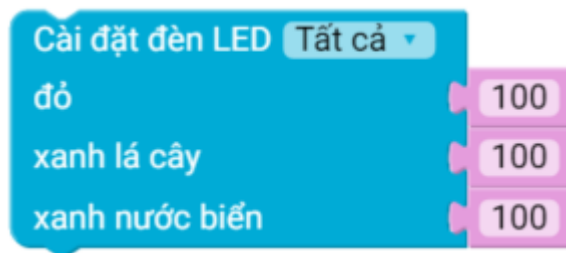
Là khối lệnh được thiết kế chuyên dụng cho phiên bản robot giáo dục KC-Bot của CTCP Phát triển Giáo dục Kidscodex.

10.1. Câu lệnh: Điều khiển động cơ

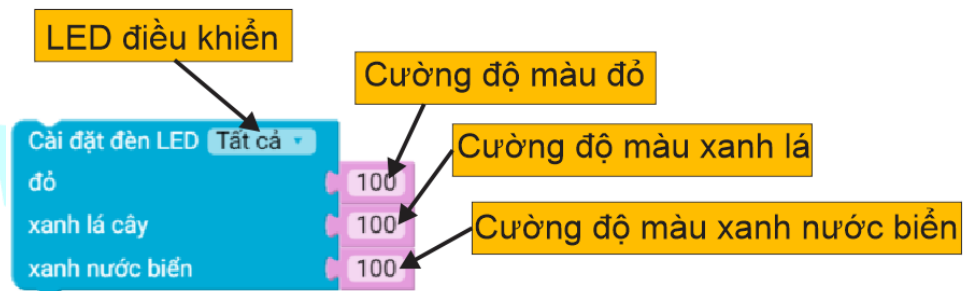
Câu lệnh	Ý nghĩa
----------	---------

Cài đặt động cơ M1 tốc độ 100	Câu lệnh điều khiển từng động cơ riêng biệt. Chúng ta điều khiển được hai cổng M1 và M2, với tốc độ tùy ý (tốc độ chạy từ 0 đến 255)
Tiến lên với nguồn 100	Câu lệnh điều khiển robot di chuyển theo các hướng tiến lên, lùi lại, quay trái, quay phải với tốc độ tùy ý (tốc độ chạy từ 0 đến 255)
Cài đặt servo S1 độ 90°	Câu lệnh điều khiển động cơ servo theo các góc xác định các cổng có thể điều khiển trên mạch là S1, S2

10.2. Câu lệnh điều khiển đèn LED RGB








Ý nghĩa: Câu lệnh điều khiển module đèn LED chuyên dụng theo bộ robot giáo dục. Số LED điều khiển là 7 trên module.



10.3. Câu lệnh đọc tín hiệu đầu vào

Câu lệnh để đọc tín hiệu từ các cảm biến trong bộ robot giáo dục KC-Bot.

Câu lệnh	Ý nghĩa
IR	Câu lệnh đọc tín hiệu từ mắt thu hồng ngoại.

 Kênh K1 phím Lên	Câu lệnh dùng để so sánh các nút đang được bấm từ điều khiển hồng ngoại.
 Đọc khoảng cách chân phát 2 chân nhận 12	Câu lệnh đọc giá trị khoảng cách từ cảm biến siêu âm HC-SR04.
 Đọc nhiệt độ (LM35) A0	Câu lệnh để đọc giá trị cảm biến nhiệt độ LM35
 Đọc nhiệt độ (ds18b20) 0	Câu lệnh để đọc giá trị cảm biến nhiệt độ DS18b20
 Đọc độ ẩm (DHT11) 0	Câu lệnh đọc giá trị từ cảm biến nhiệt độ độ ẩm DHT11. Chúng ta có thể đọc được 2 thông số nhiệt độ và độ ẩm

10.4. Câu lệnh thời gian

Câu lệnh khởi tạo thời gian.

Cài đặt thời gian 1 : 1 : 1 ngày trong tuần Chủ nhật 1 - 1 - 2018

Ý nghĩa: Câu lệnh khởi tạo thời gian cho module thời gian thực DS1307.

Câu lệnh đọc thời gian.

Đọc thời gian rtc Giờ

Ý nghĩa: Câu lệnh đọc về giá trị thời gian từ module thời gian thực

11. Câu lệnh cho robot giáo dục Arduino robot.

Khối lệnh được tạo ra chuyên dụng để sử dụng cho phiên bản robot giáo dục Arduino robot.

12. Câu lệnh lập trình những hệ thống IOT.

12.1. Câu lệnh: Kết nối đến một địa chỉ Wifi.

Kết nối Wifi tên " " mật khẩu " "

Ý nghĩa: Kết nối đến một địa chỉ wifi mong muốn.

Ví dụ:

Kết nối Wifi tên " KIDSCODE " mật khẩu " KIDSCODE123 "

12.2. Câu lệnh: Gửi dữ liệu lên một kênh trên Thingspeak.

Gửi dữ liệu 0 lên biểu đồ 1 với WirtAPIKey " "

Ý nghĩa: Gửi dữ liệu lên biểu đồ với địa chỉ xác định trên trang thingspeak

Ví dụ:

Gửi dữ liệu lên biểu đồ với WirtAPIKey

LỜI KẾT

Với phần mềm lập trình kéo thả EasyCode chúng ta có thể dễ dàng viết chương trình lập trình cho các board mạch phổ biến. Tài liệu này sẽ hướng dẫn các bạn những kiến thức có bản để có thể bắt đầu lập trình với phần mềm.