***PHẦN NÂNG CAO***

*Trong phần này ta sẽ kết hợp kết quả các bài toán trước để giải những vấn đề phức tạp hơn như đi trong mô hình bàn cờ, đi vào mê cung.*

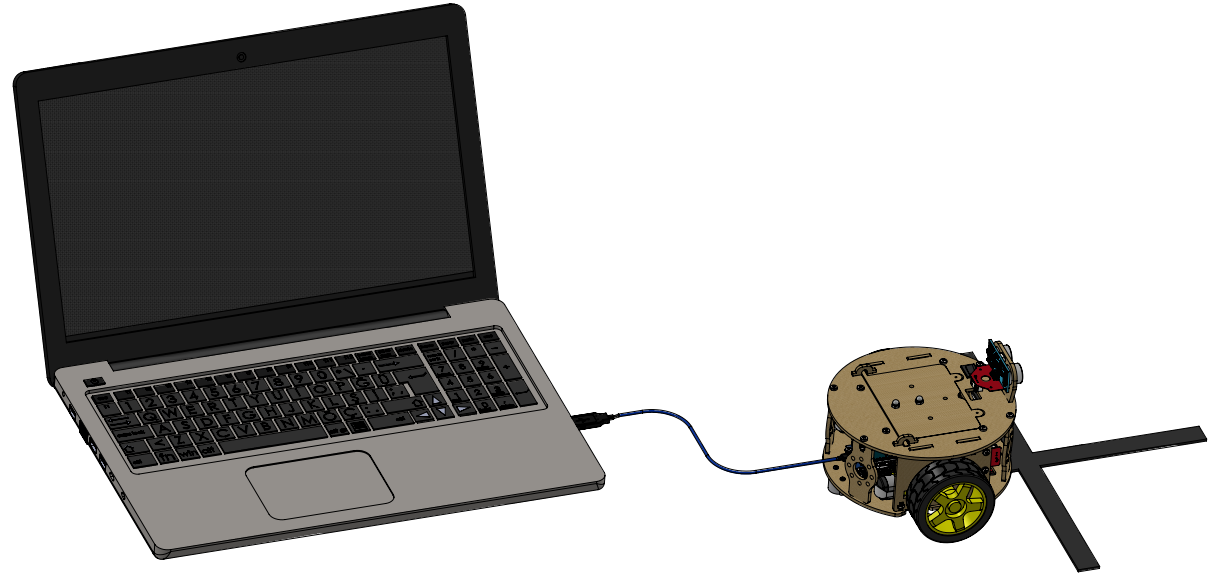
**Bài tập 13:**  Xác định và đếm sô ngã tư

- Phương pháp Xác định đi qua ngã tư để làm cơ sở đếm số ngã tư đã đi qua

- Viết chương trình đếm số ngã tư và truyền lên máy tính để kiểm tra

**Kiến thức cần nắm**

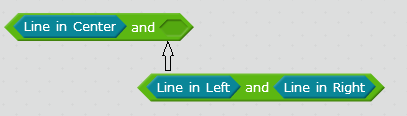
|  |  |  |
| --- | --- | --- |
| Blocks | Cách sử dụng | Ví dụ |
| Tạo một biến lưu trữ dữ liệu | Tạo ra một biến (là đơn vị lưu trữ dữ liệu) trung gian để giải quyết vấn đề gì đó ví dụ như số ngã tư đã đi qua, | Tạo ra biến Count để lưu trữ số đếm. |
| operators “and”  Toán tử “and” | Toán tử **and** trong nhóm operators sử dụng để kiểm tra các điều kiện xảy ra đồng thời. Nếu các điều kiện cùng đúng thì nó trả về kết quả là đúng, nếu chỉ cần một điều kiện sai thì nó sẽ tra về kết quả là sai. | Nếu cảm biến bên phải và cảm biến bên trái đều đồng thời phát hiện vạch đen thì trả về kết quả là đúng, nếu một trong hai không phát hiện vạch đen thì trả về kết quả là sai |
|  |  |  |
|  |  |  |



Chuẩn bị: kết nối robot với máy tính và phần mềm mBlock, Robot đặt trên 1 tờ giấy in hình vạch ngã tư, chúng ta sẽ đẩy robot qua ngã tư để kiểm tra số liệu đếm ngã tư được gửi lên máy tính có đúng hay không.

Phương pháp xác định ngã tư: *Ngã tư được xác định khi cả 3 mắt đều phát hiện vạch kẻ.*

Có 1 vấn đề: Toán tử **and**  chỉ kiểm tra 2 điều kiện, làm thế nào để kiểm tra 3 điều kiện? ta sử dụng 2 toán tử  cách làm như sau:

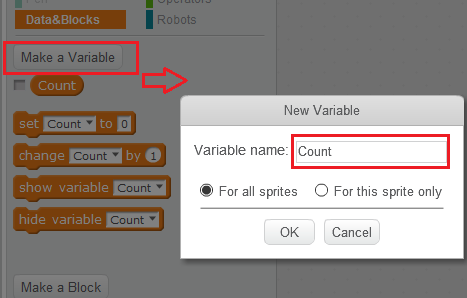


Như vậy ta sẽ có được phép kiểm tra **And** với 3 điều kiện: 

***Nếu phép kiểm tra này là đúng thì có nghĩa là 1 ngã tư đã được phát hiện***

Khi kiểm tra được cả 3 cảm biến đều nhận vạch kẻ đen thì ta tăng một số đếm trung gian lên 1 đơn vị và gửi số đó lên máy tính. Muốn vậy ta phải tạo ra 1 biến (variable) để chứa giá trị số đếm.

Trong nhóm block **Data&Blocks** , bấm vào “**Make a Variable”,** sau đó gõ tên biến là Count



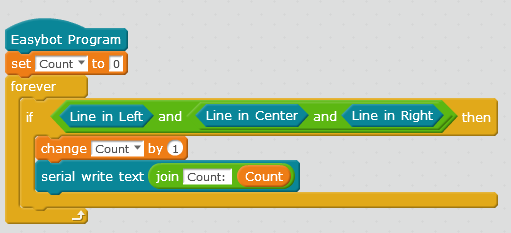
Ta đã tạo 1 biến Count. Các thao tác với biến Count gồm có:

: block chứa giá trị biến Count

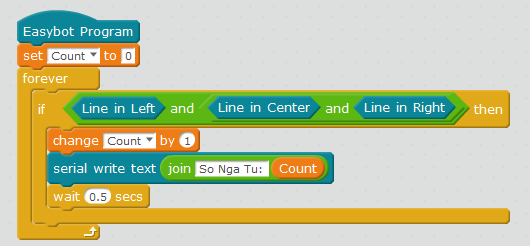
 đặt giá trị cho biến Count: về toán học nó giống như gán: Count = n

 thay đổi giá trị biến count bằng cách tăng hay giảm 1 lượng. Về toán học nó giống với count = count + n (với n>0: tăng, với n<0: giảm)

Ta hoàn thiện chương trình như sau: nếu cả 3 cảm biến đều phát hiện vạch đen thì tăng Count lên 1 đơn vị. Nạp chương trình và đẩy robot đi ngang ngã tư để quan sát dữ liệu gửi lên máy tính.



Chương trình trên hoạt động được nhưng sẽ có 1 vấn đề, số biến đếm tăng nhiều khi cho robot lướt qua 1 vạch ngang. Lý do là do tốc độ thực thi chương trình rất nhanh so với tốc độ di chuyển của Robot do đó khi Robot còn đang ở trên vạch ngang chưa ra khỏi vạch thì chương trình tiếp tục lặp lại đếm và nó lại tăng biến đếm -> không chính xác. Giải pháp là sử dụng lệnh đợi 1 thời gian trước khi lặp lại việc đếm vạch:



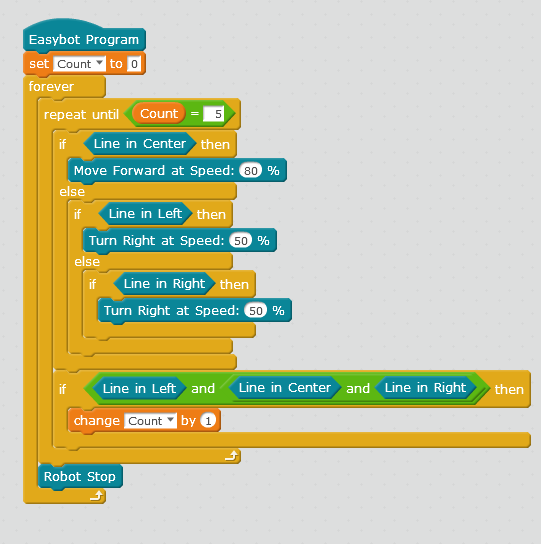
Thời gian chờ chú ý lựa chọn phù hợp với tốc độ di chuyển của robot để đảm bảo không bị trễ việc cho các tác vụ khác.

**Bài tập 14**: *Viết chương trình kết hợp dò đường theo vạch đen và đếm số ngã tư, khi đến vị trí yêu cầu (xác định bằng số ngã tư đã đi qua) thì dừng lại.*

**Kiến thức cần nắm:**

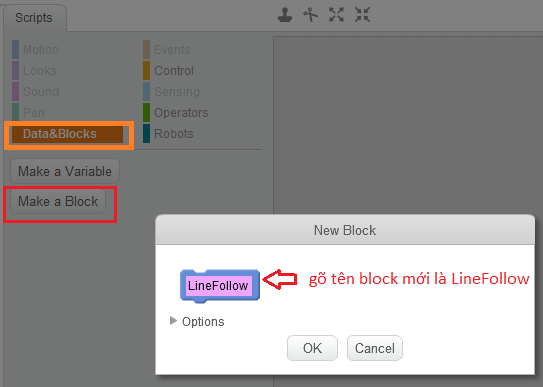
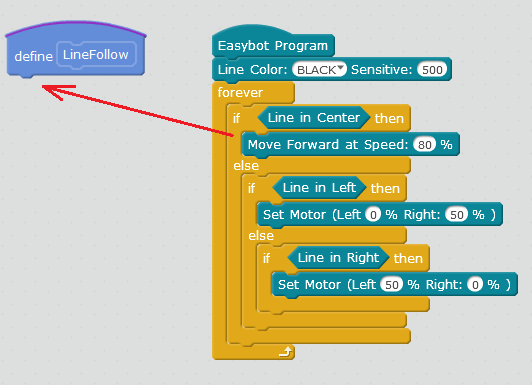
|  |  |  |
| --- | --- | --- |
| **Blocks** | **Giải thích** | **Ví dụ** |
| Tạo một block riêng từ nhiều block khác nhau | Khi viết một chương trình phức tạp để giải quyết vấn đề, ta có thể làm gọn chưa trình bằng cách tạo ra 1 block mới chưa các đoạn mã block mà có nhu cầu sử dụng lặp đi lặp lại nhiều lần. | Tạo block dò đường, và sau này ta có thể tái sử dụng để kết hợp vừa dò đường vừa làm việc khác một cách dễ dàng và mạch lạc |
|  |  |  |

Gợi ý: Kết hợp chương trình dò vạch ở **bài tập 11** và phương pháp đếm ngã tư ở **bài tập 13**, phối hợp trong cùng 1 chương trình Scratch như thế này:

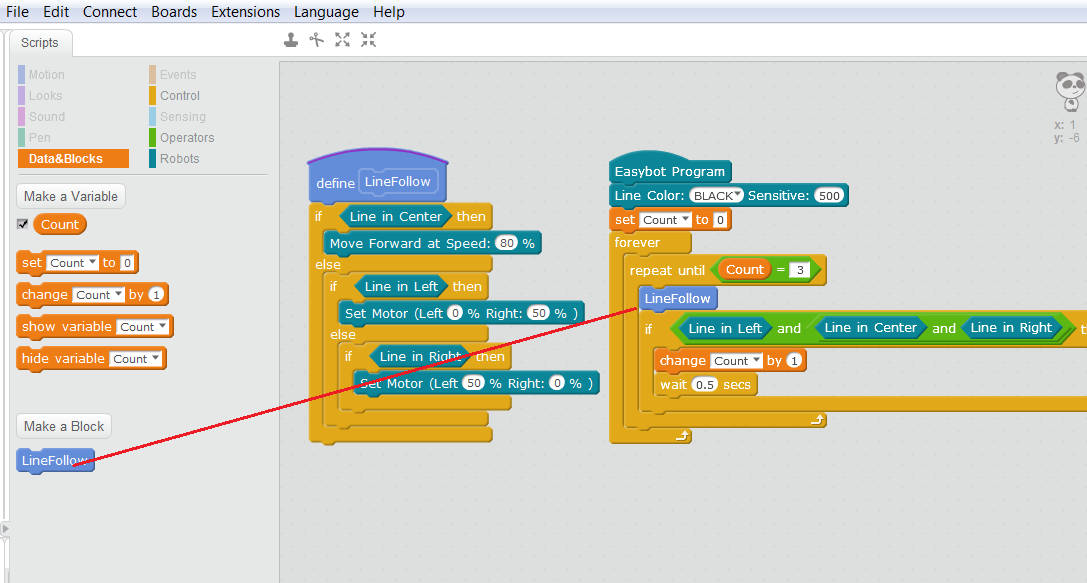


Lúc này chương trình có vẻ khá phức tạp và khó quản lý khi các bạn muốn có một số sửa đổi cải tiến về chương trinh ?

Đừng lo có 1 giải pháp khác đó là tạo ra 1 Block mới đóng vài trò như 1 chương trình con. Ta sẽ gộp tất cả những đoạn mã liên quan đến dò line và 1 block để chương trình dễ quản lý hơn và dễ dàng triển khai phối hợp với các chương trình khác kết hợp dò line.

Thực hiện như sau: vào group **Data&Blocks** bấm vào Make a Block  

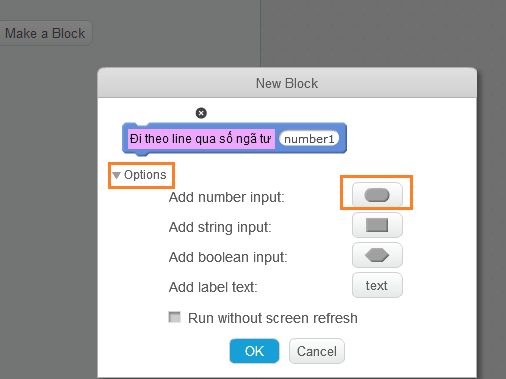
Sau đó kéo và thả cả đoạn chương trình liên quan đến dò line vào Block mới tạo



Giờ thì chương trình trông rất khoa học và bài bản, nếu muốn sửa đổi nâng cấp các thuật toán dò đường các bạn có thể làm tự do mà không sợ bị ảnh hưởng đến cấu trúc chương trình chính.

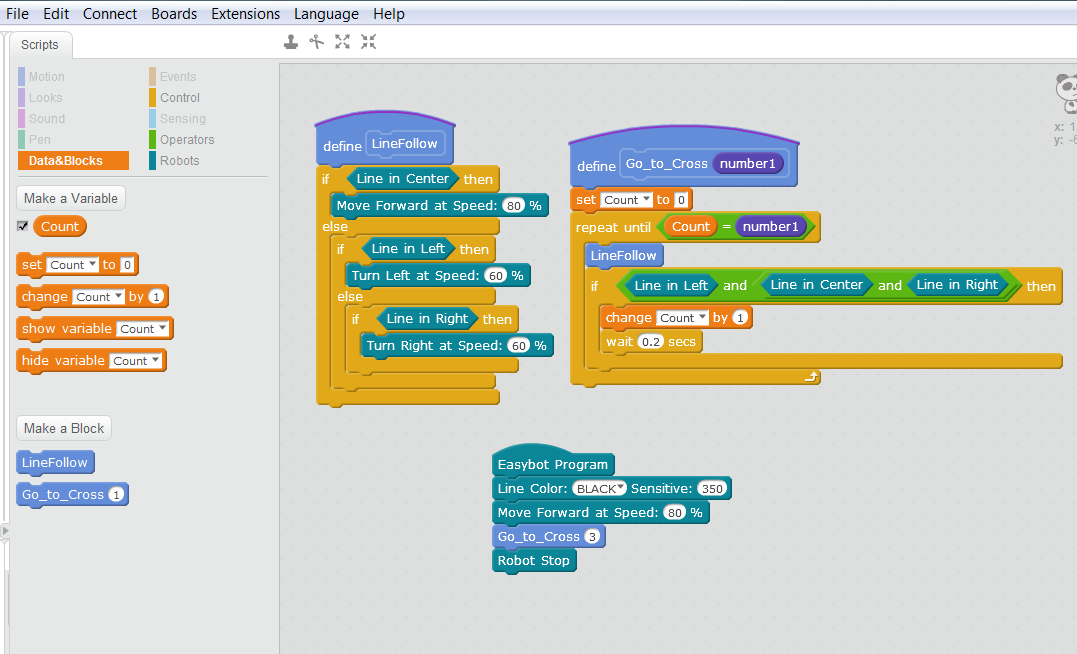
**Bài tập 15:**  *Tương tự bài tập 14 nhưng triển khai theo cách khoa học hơn,* ***Hãy tạo ra một block mới*** *(chương trình con) thực hiện chức năng vừa dò đường và đếm đủ số ngã tư thì dừng. Tham số đầu vào (input) của Block là số ngã tư phải vượt qua.*

Gợi ý: Khi tạo block mới, vào option, ta thêm số liệu đầu vào (add number input) để nhập số ngã tư cần vượt qua

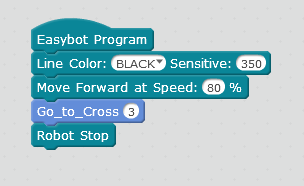


Sau đó tiếp tục kế thừa bài tập 14, vẫn tiếp tục sử dụng block Linefollow và tạo thêm block Go\_to\_Cross (number) để di chuyển vừa sử dụng block linefollow vừa đếm số ngã tư.

Như vậy ta đã tạo ra 1 block rất hữu ích cho các chương trình đi vào sa hình dò line phức tạp.



Như vậy Chương trình chính trở nên rất đơn giản, ví dụ muốn dò line qua 3 ngã tư thì dừng:



Với bài tập này, các bạn sẽ thành thạo trong việc tạo ra các block chức năng riêng từ những block sẵn có để tái sử dụng cho các chương trình có độ phức tạp cao hơn.

**Bài tập 16:** *Viết chương trình tạo block con quay phải 90 độ quanh ngã tư.*

*Chú ý, đề bài ngắn nhưng khó ở chỗ phải đảm bảo robot quay ra khỏi vạch đen rồi mới kiểm tra đã quay đủ góc ngẵ từ hay chưa. Ứng dụng Block vừa tạo để quay 4 góc của 1 ngã tư.*

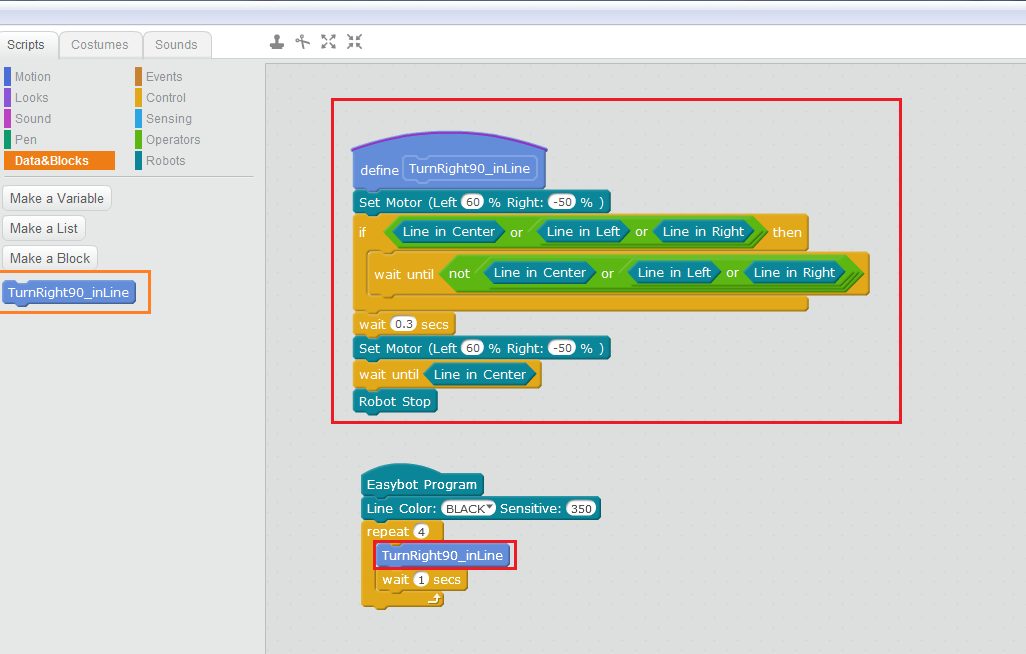
*Suy nghĩ như sau:*

*---Khi robot đang còn ở trên vạch*

*--------Quay phải cho đến khi không còn vạch*

*--Delay 1 khoảng 0.2s để thực sự ra khỏi vạch*

*---Tiếp tục quay phải cho đến khi cảm biến ở giữa nhận được vạch*

**

*Như vậy chúng ta đã tạo ra được các block con về đi theo vạch và đếm ngã tư, rẽ trái / rẽ phải ở Ngã tư.*