

РОБОТОТЕХНИКА

Инженерно-технические кадры инновационной России



Данная презентация может быть использована как на уроках информатики и технологии, так и во внеурочной деятельности, на кружке «Робототехника», во время подготовки к соревнованиям.

Презентация рассчитана на учащихся 9-11 классов

Робот для траектории

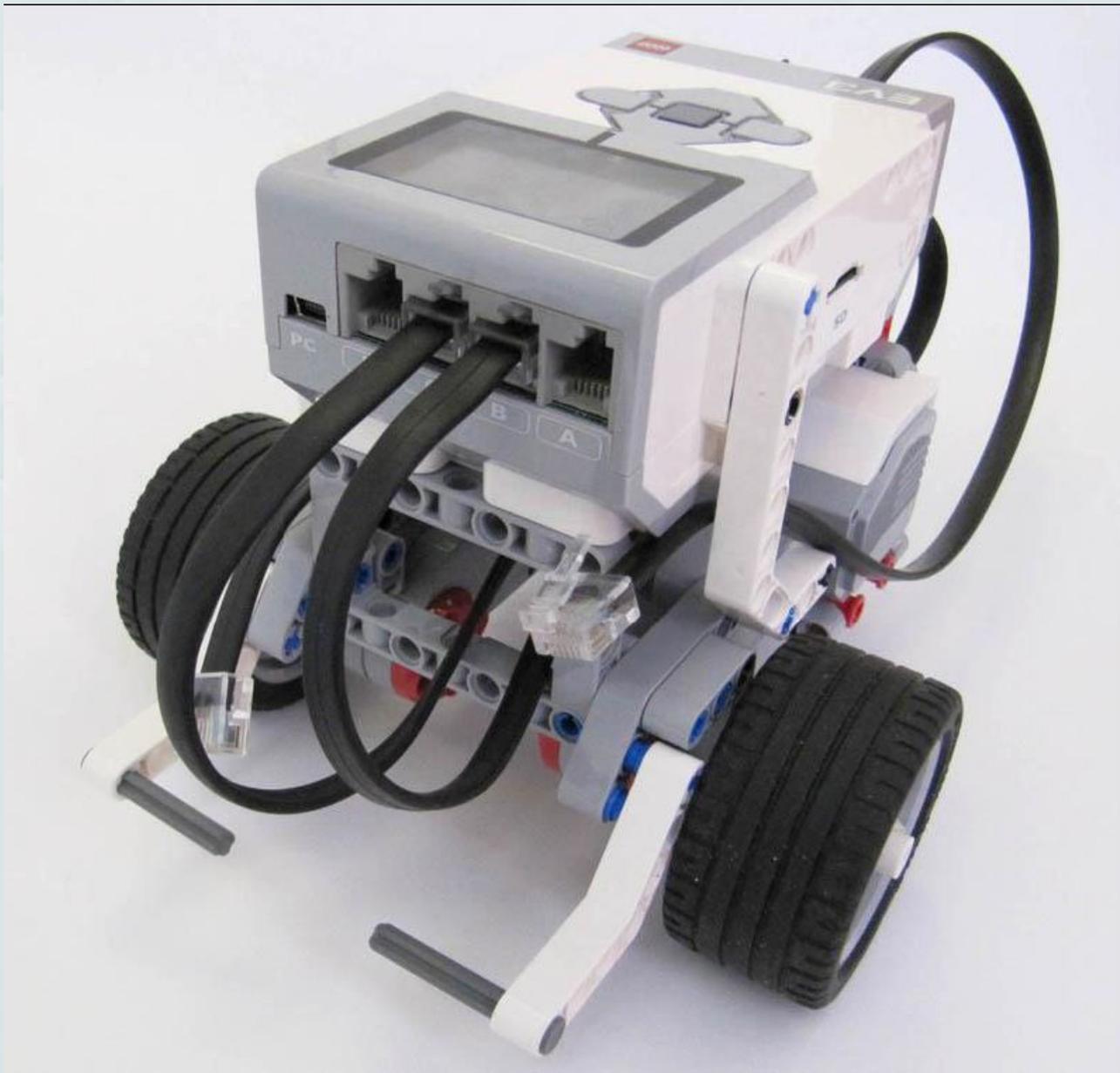
Lego Mindstorms EV3



Собираем базовую модель учебного робота Robot Educator

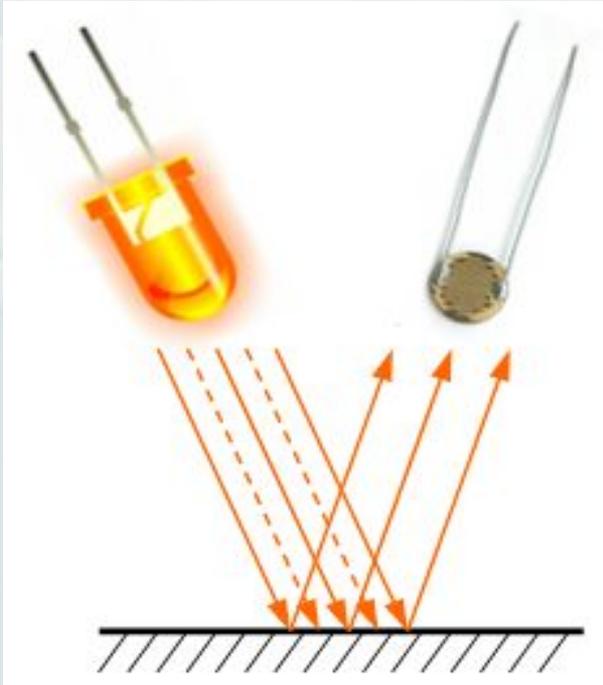
Для этого можно использовать инструкцию в программном обеспечении MINDSTORMS EV3.





Так же нам понадобятся,
датчики света-цвета EV3.





В режиме измерения окружающей освещенности, количество света, попавшее на светочувствительный элемент, преобразуется в цифровое значение, которое уже используется в программе.

В режиме измерения отраженного цвета, помимо светочувствительного элемента, активируется светоиспускающий элемент (светодиод). Свет, выпущенный этим элементом, отражается от какой-нибудь поверхности и попадает обратно в светочувствительный элемент.

Что выполняет датчик освещенности?

Измеряет уровень освещенности.

Например, показания сенсора может быть 10% на темной стороне и 90% процентов на светлой.



На какой высоте от поверхности поля и под каким углом лучше всего крепить датчик освещенности?

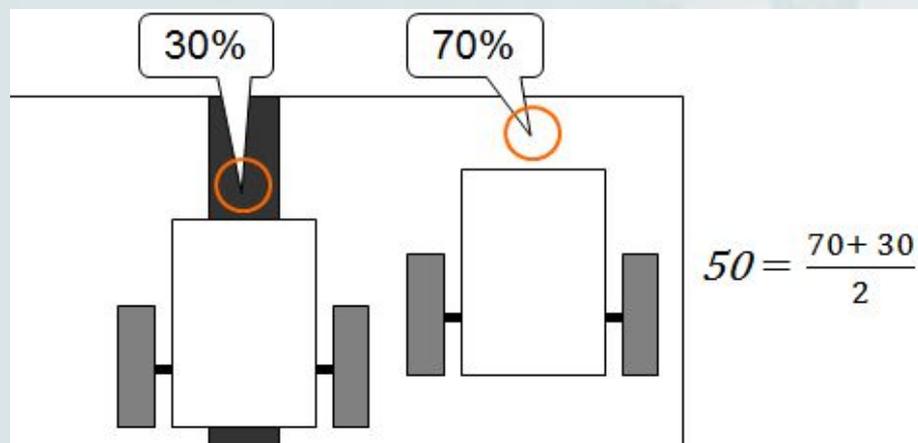
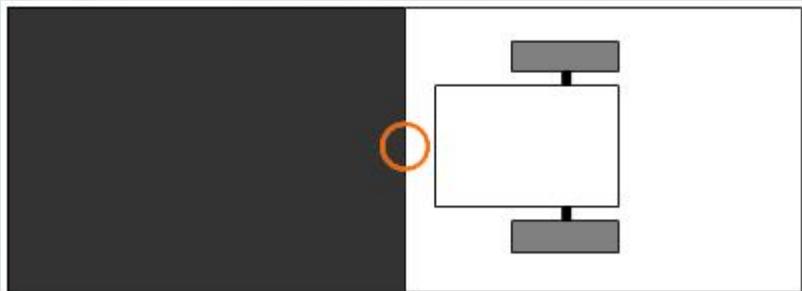


- На расстоянии 1 см. и менее от поверхности
- Под углом 90° и менее относительно робота



Что такое средняя освещенность и по какой формуле она рассчитывается?

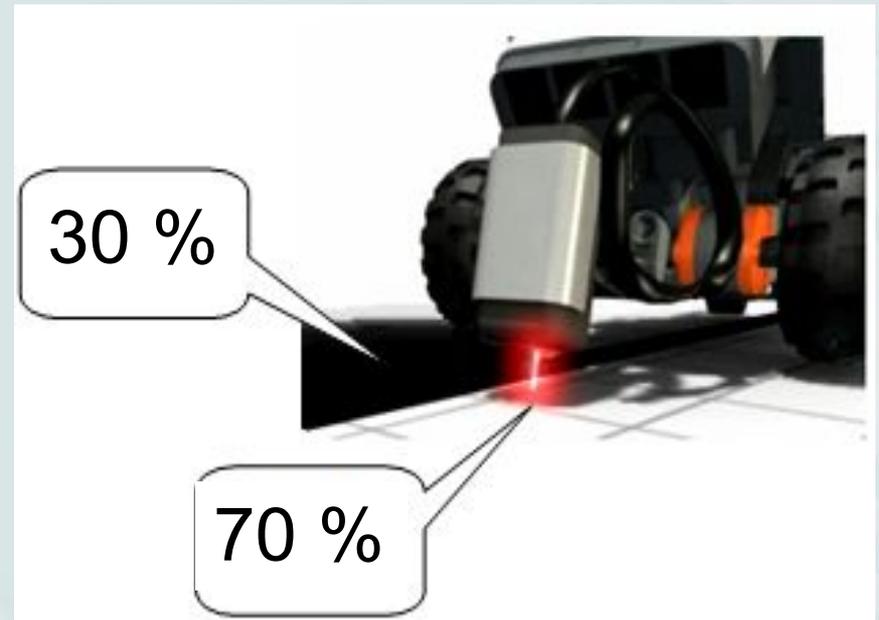
Это освещенность на границе



$$I_{\text{ср}} = (I_{\text{св}} + I_{\text{тем}}) / 2$$

Перед началом программирования, необходимо провести калибровку сенсора освещенности. После чего, измерить, что показывает сенсор на разных частях карты светлой и темной части карты. Пусть после калибровки, показания сенсора будут 30% на темной стороне и 70% процентов на светлой.

Следовательно, условием, когда можно рассматривать, что робот уже на темной стороне – значение на сенсоре стало меньше 50%.



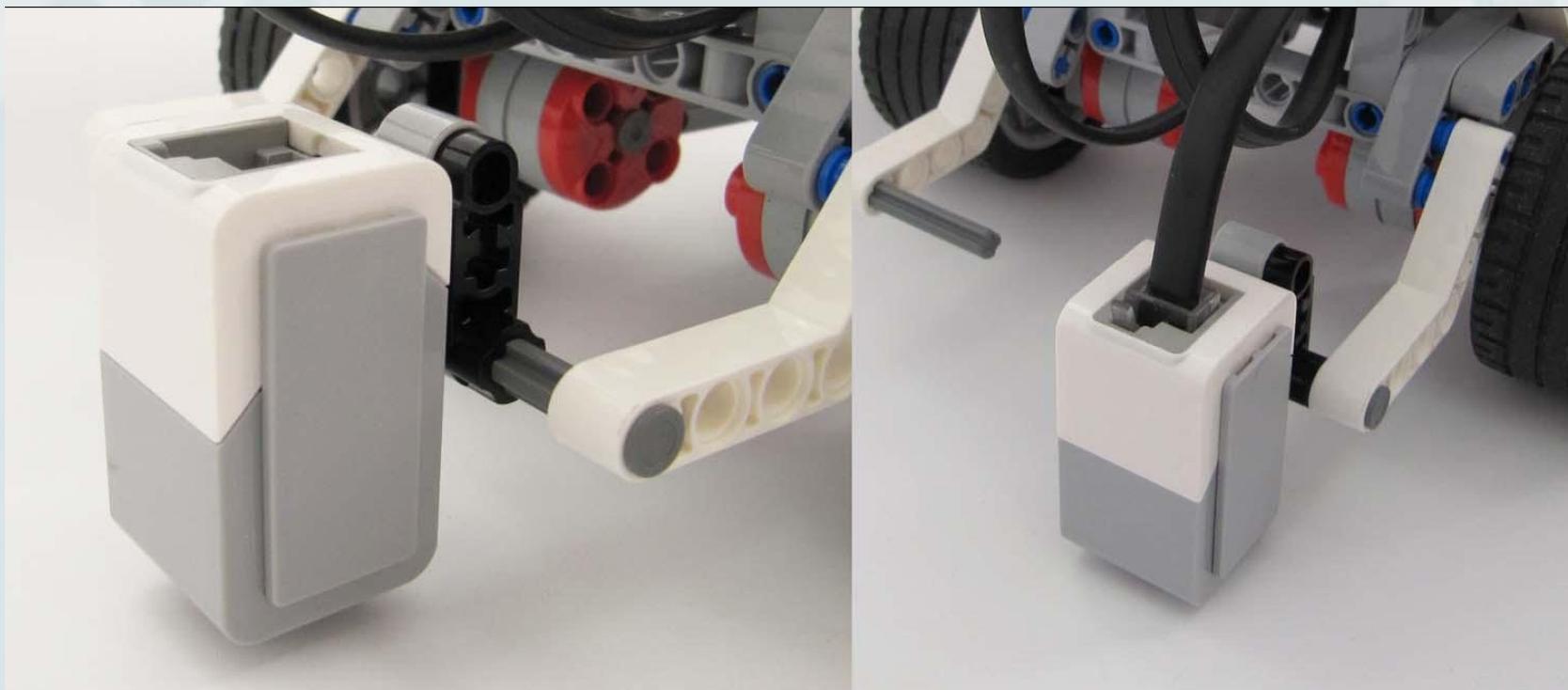
Для примера мы разберём 3 примера программ для движения по чёрной траектории изображённой на ровном, светлом фоне:

- Один датчик, с П регулятором.
- Один датчик, с ПК регулятором.
- Два датчика.

Пример 1.
Один датчик, с П регулятором.

Конструкция

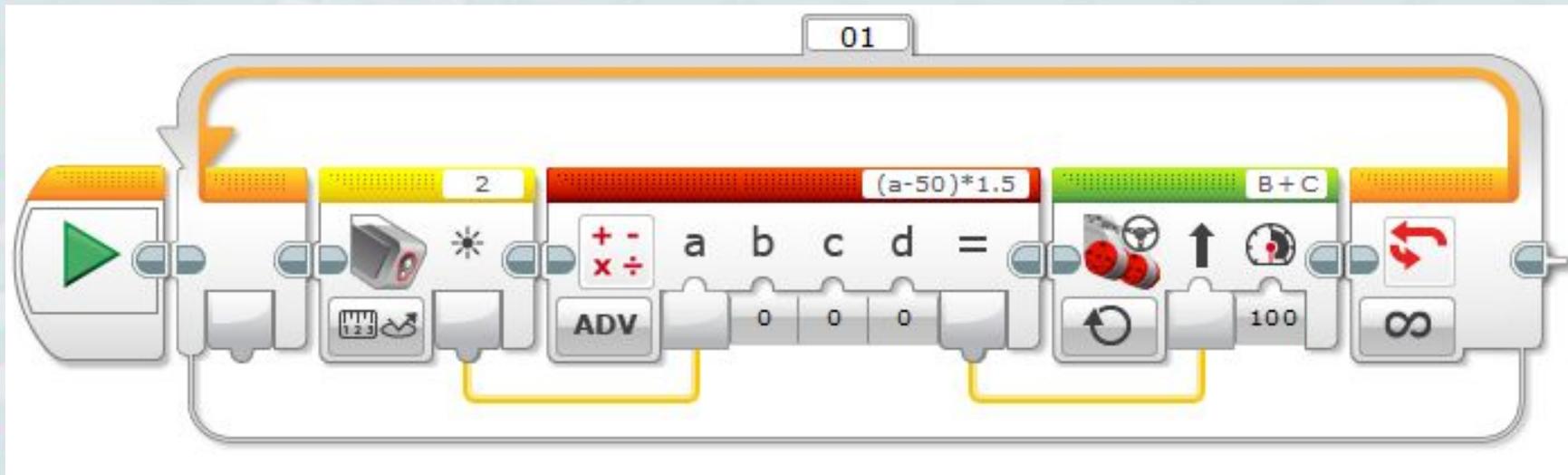
Датчик света устанавливается на балку, удобно расположенную на модели.



Алгоритм

Действие алгоритма основано на том, что в зависимости от степени перекрытия, пучка подсветки датчика чёрной линией, возвращаемые датчиком показания градиентно варьируются. Робот сохраняет положение датчика света на границе чёрной линии. Преобразовывая входные данные от датчика света, система управления формирует значение скорости поворота робота.

Так как на реальной траектории датчик формирует значения во всём своём рабочем диапазоне (0-100), то значением к которому стремиться робот, выбрано 50. В этом случае значения передаваемые функции поворота формируются в диапазоне -50 - 50, но этих значений недостаточно для крутого поворота траектории. По этому следует расширить диапазон в полтора раза до -75 - 75.



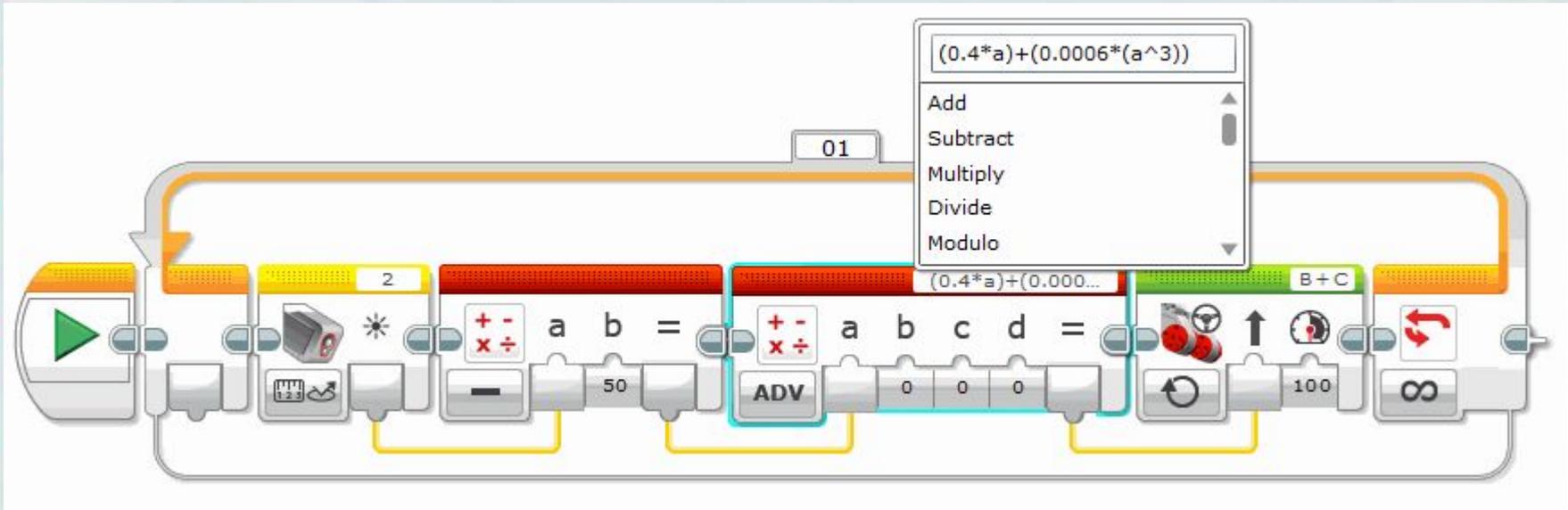
Более устойчиво алгоритм работает, если использовать моторы с управлением скоростью $-100...100$.

В этом случае есть возможность отрегулировать плавность поворота в соответствии с кривизной линии

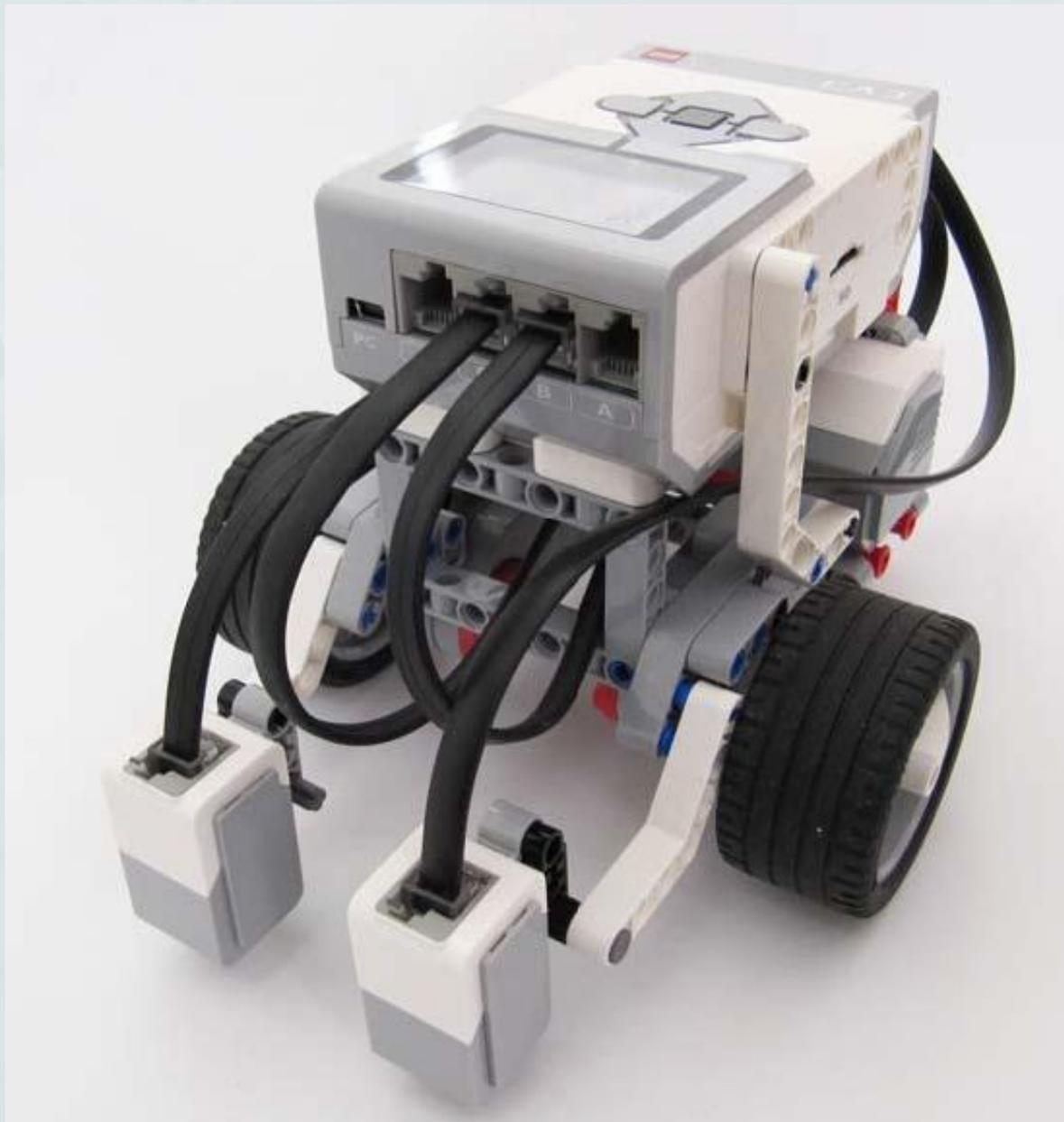
Так как на реальной траектории датчик формирует значения во всём своём рабочем диапазоне (0-100), то значением к которому стремиться робот, выбрано 50. В этом случае значения передаваемые функции поворота формируются в диапазоне -50 - 50, но этих значений недостаточно для крутого поворота траектории. По этому следует расширить диапазон в полтора раза до -75 - 75.

Пример 2. Один датчик, с ПК регулятором.

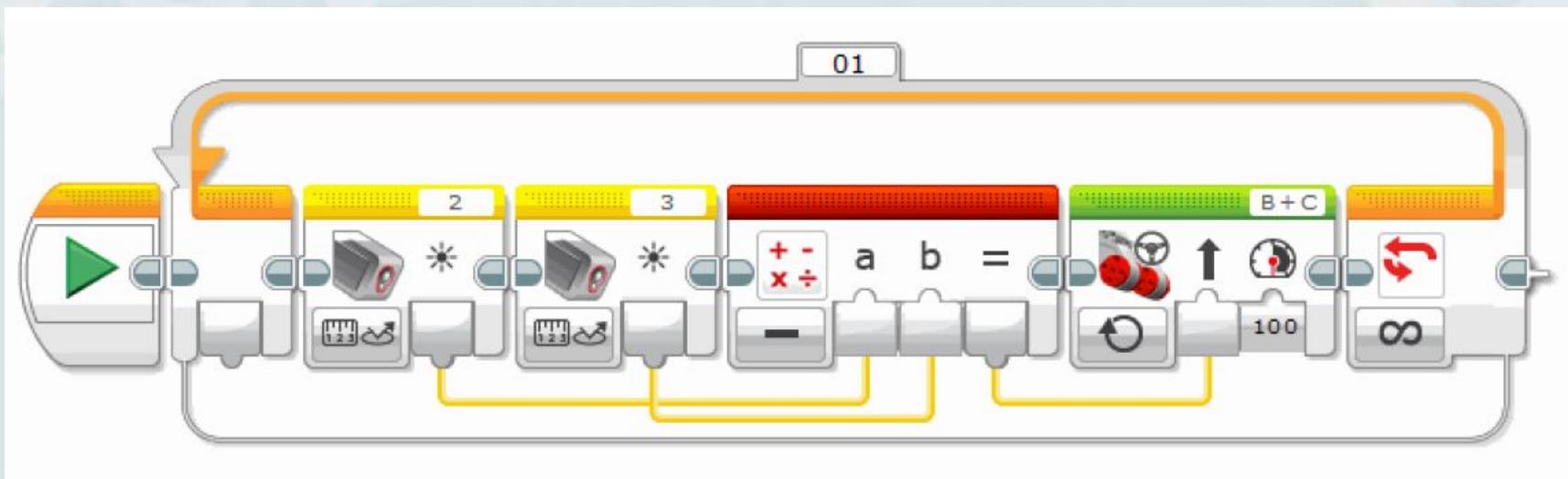
К нашему пропорциональному регулятору мы добавляем еще и простой кубический регулятор, который добавит изгиб в функции регулятора. Это позволит уменьшить раскачивание робота рядом с нужной границей траектории, а так же совершать более сильные рывки при сильном удалении от неё



Пример 3. Два датчика



Использование двух датчиков позволяет более чётко разграничить отклонение датчиков от линии и позволяет легко отфильтровывать/подсчитывать перекрёстки или сложные повороты на траектории.



Возможные проблемы

Робот крутится на месте, не заезжая на линию

В этом случае следует либо стартовать с другой стороны линии, либо поменять подключения моторов к контроллеру местами.

Робот проскакивает линию, не успевая среагировать.

Следует понизить мощность моторов.

Робот реагирует на мелкие помехи на белом, не доезжая до черного.

Надо увеличить порог чувствительности датчика (например, не на 5, а на 8 пунктов). Вообще говоря, это число можно рассчитать. Для этого следует снять показания датчика на белом, затем на черном, вычесть одно из другого и поделить пополам. Например, $(56 - 40) / 2 = 8$.

Список источников

- Овсянницкая Л.Ю. Курс программирования робота Lego Mindstorms EV3 в среде EV3: основные подходы, практические примеры, секреты мастерства. – Челябинск: ИП Мякотин И.В., 2014
- Вязовов С.М., Калягина О.Ю., Слезин К.А. Соревновательная робототехника: приемы программирования в среде EV3. – М.: Издательство «Перо», 2014
- <http://robot.edu54.ru/>
- <http://pilotlz.ru/robo/>