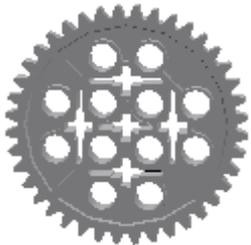

Регуляторы для движения по границе черного и белого в TRIK Studio



Президентский физико-математический лицей № 239
Сергей Александрович Филиппов

П л а н з а н я т и я

- Регуляторы
- Ветвление
- Задача движения по границе
- Значение серого
- Релейные регуляторы
- Двухпозиционный регулятор
- Калибровка датчика
- Корректировка скорости
- Трехпозиционный регулятор

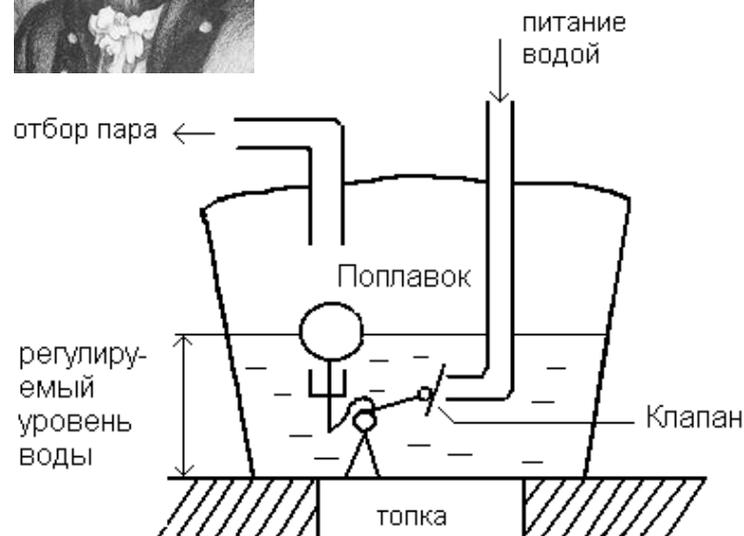


Регулятор

- Регулятор — это совокупность устройств, обеспечивающих их желаемое поведение системы.



Поплавковый регулятор
Ползунова,
1765 г.

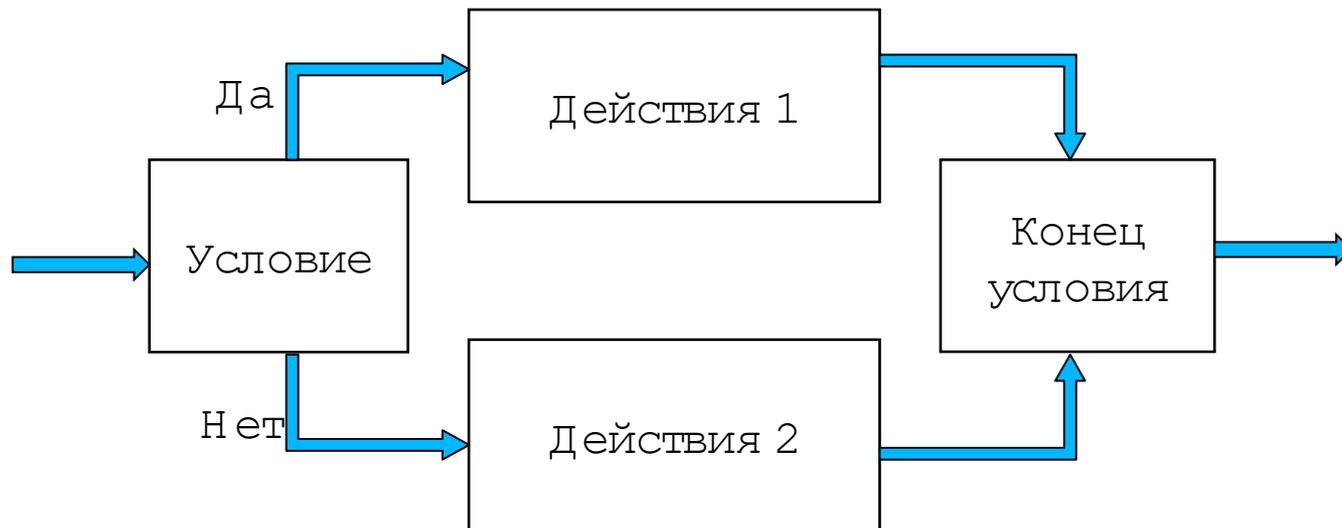


Центробежный регулятор
Уатта,
1788 г.



Ветвление

- Ветвление – часть алгоритма, в которой в зависимости от условия выполняется один или другой набор действий.



Движение вдоль границы черного и белого

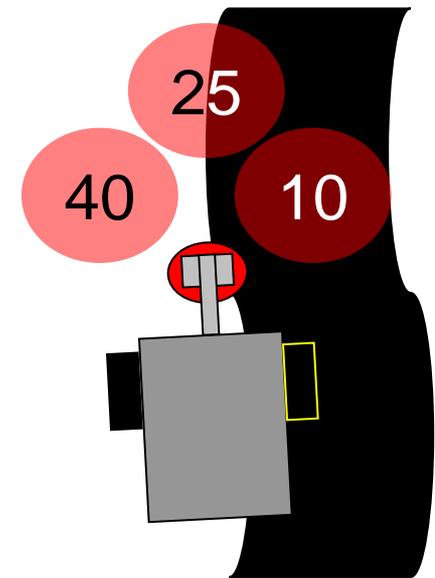
- Задача двигаться по границе черного и белого с помощью датчика освещенности
- Значение серого находится на границе и вычисляется через среднее арифметическое:

$$\text{grey} = (\text{white} + \text{black}) / 2$$

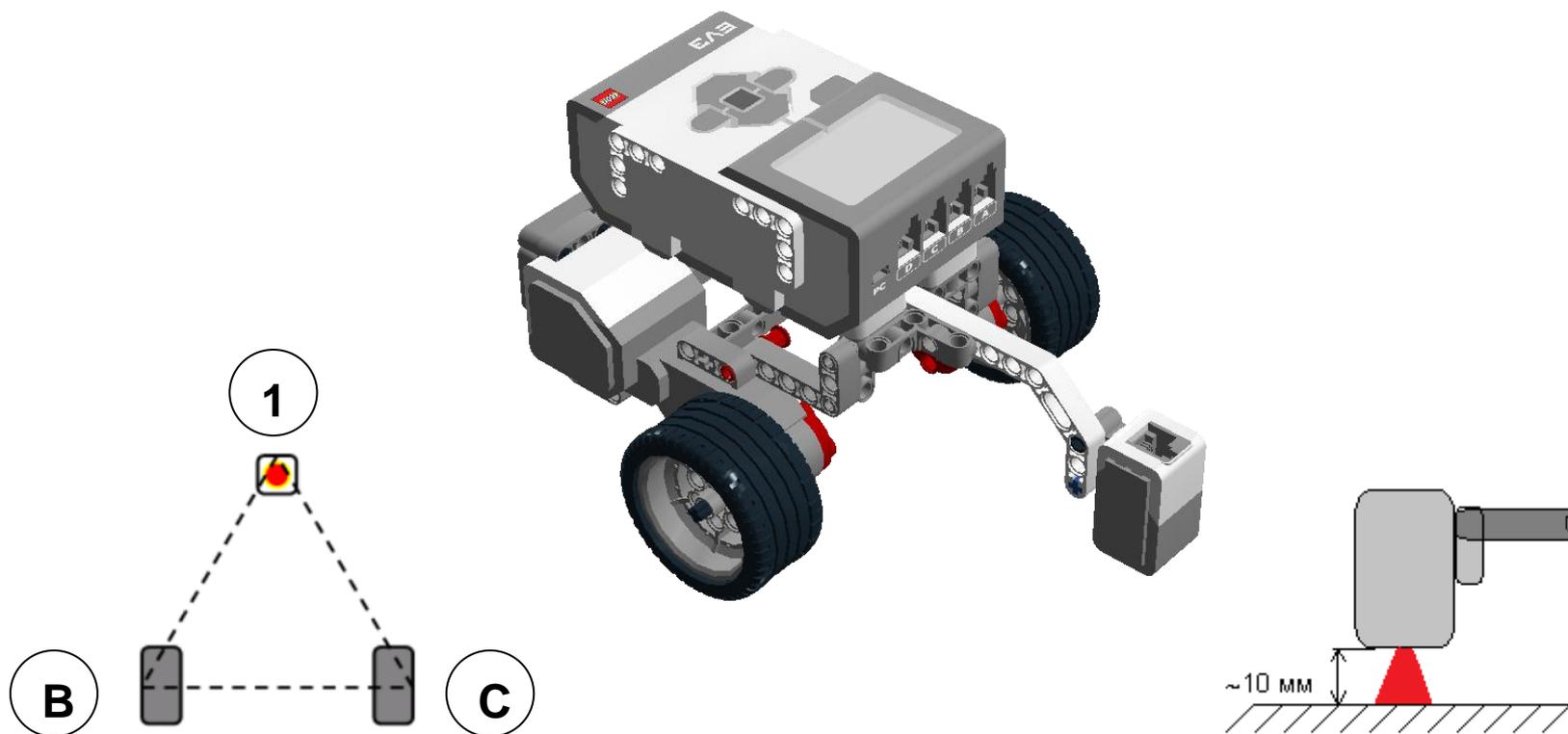
- Для каждого робота свое значение серого, например

$$\text{grey} = 25$$

- **Калибровка** – определение пороговых показаний датчика

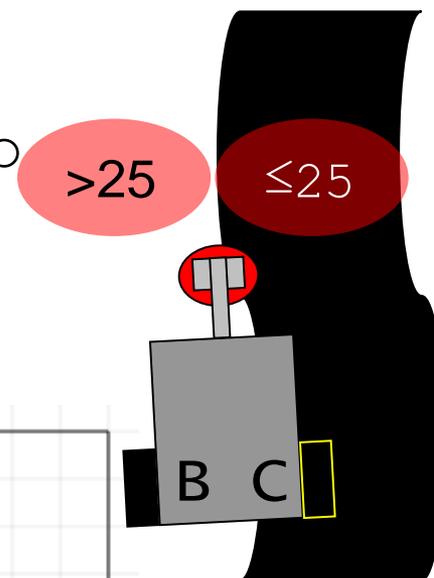
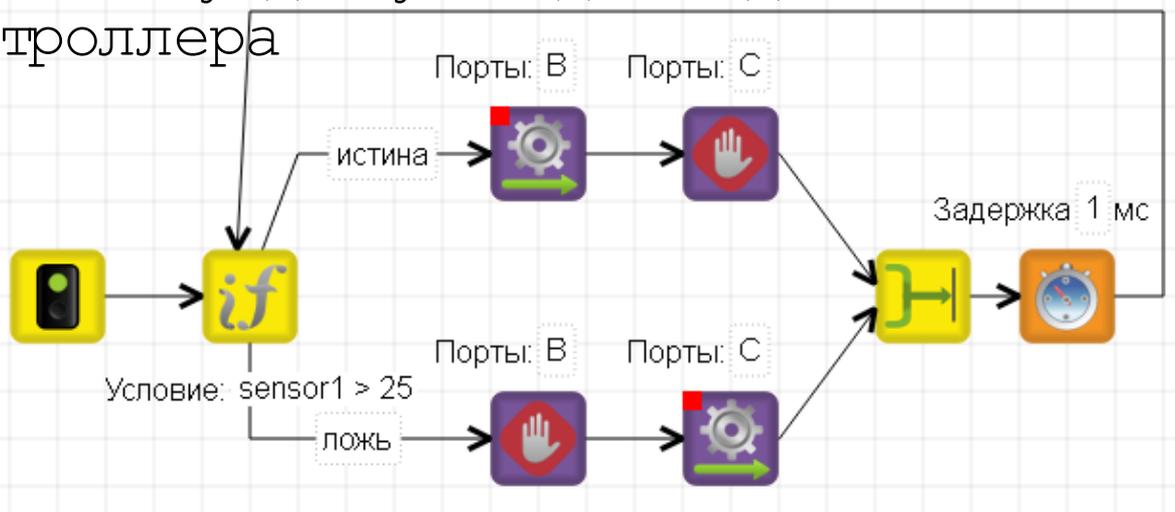


Тележка с одним датчиком освещенности



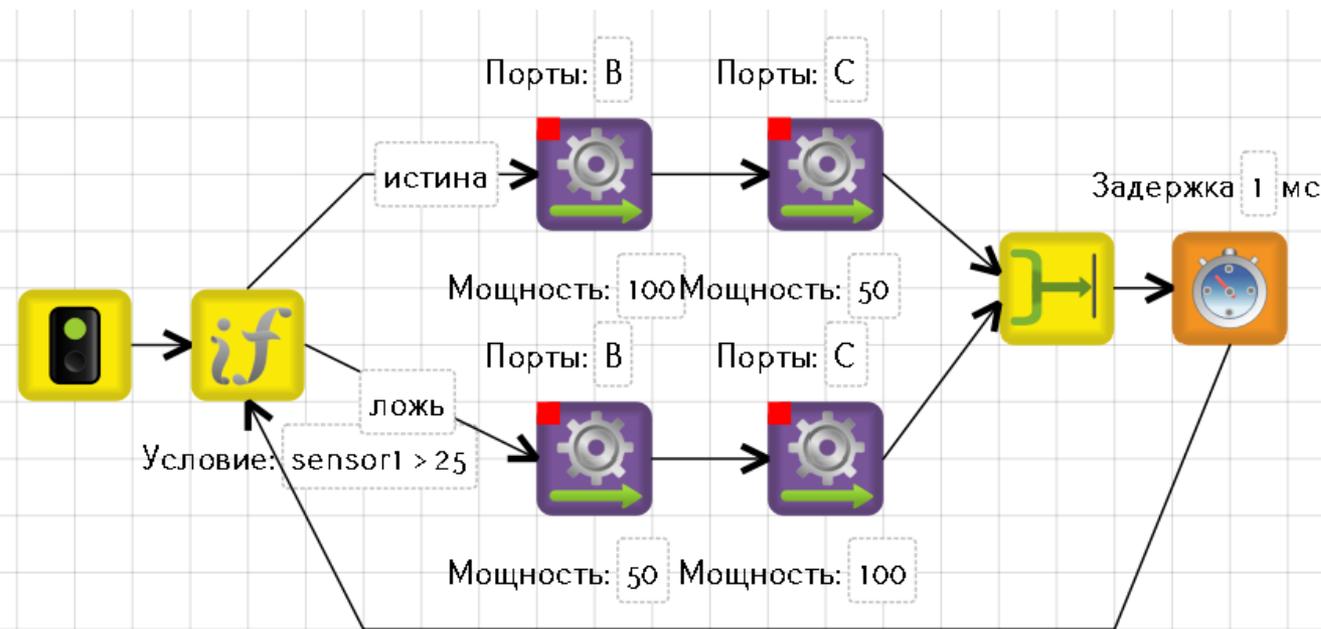
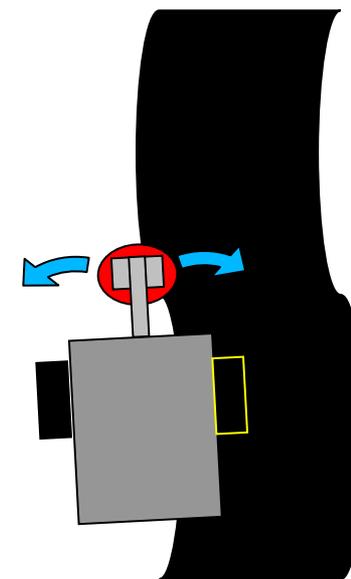
Двухпозиционный релейный регулятор

- Двухпозиционный релейный регулятор имеет два состояния:
 - освещенность $>$ серого,
 - освещенность \leq серого.
- Двухпозиционный регулятор строится на основе ветвления и выполняет одно из двух действий:
 - плавный поворот направо,
 - плавный поворот налево.
- Миллисекунда нужна для отдыха контроллера



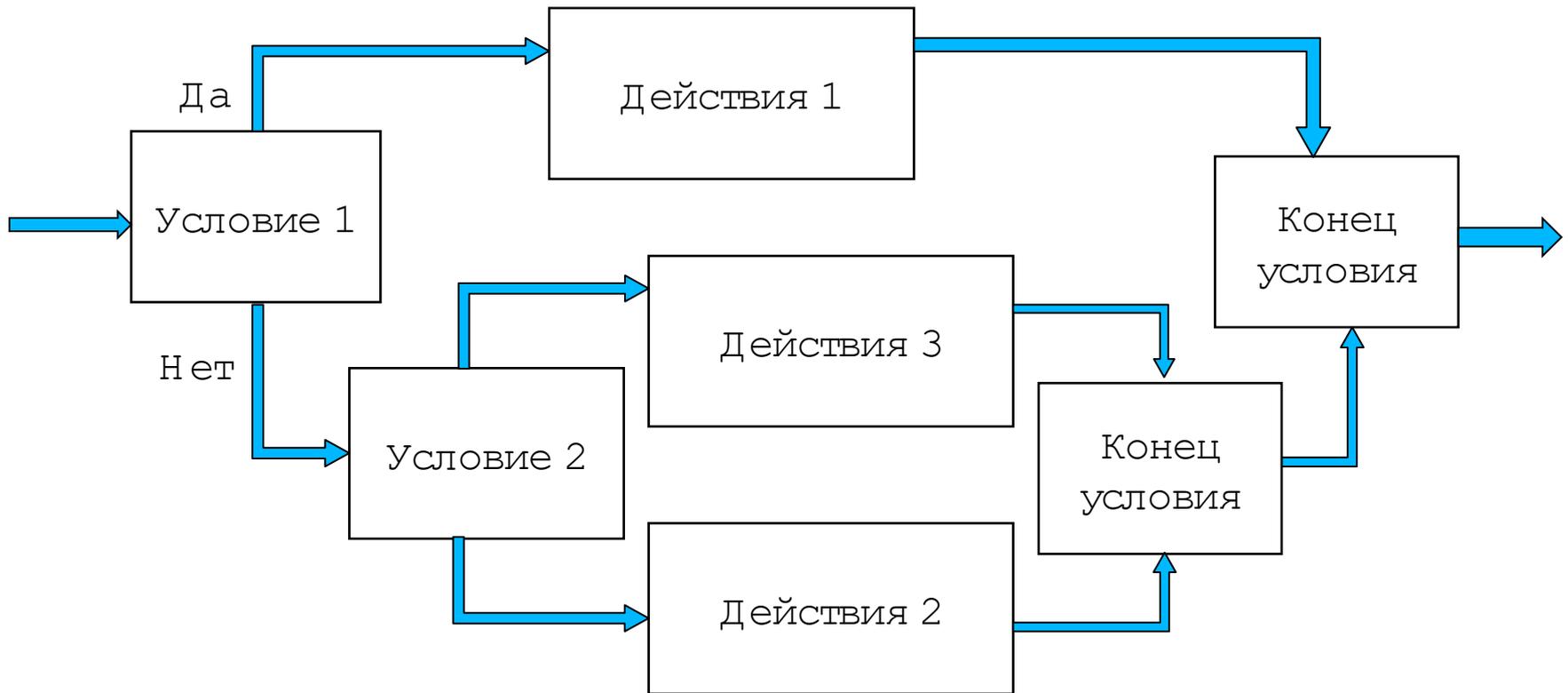
Релейный регулятор : движение вдоль границы черного и белого

- Движение на релейном регуляторе слишком неравноное : работа заносит
- Для стабилизации можно понизить скорость
- Или наоборот повысить



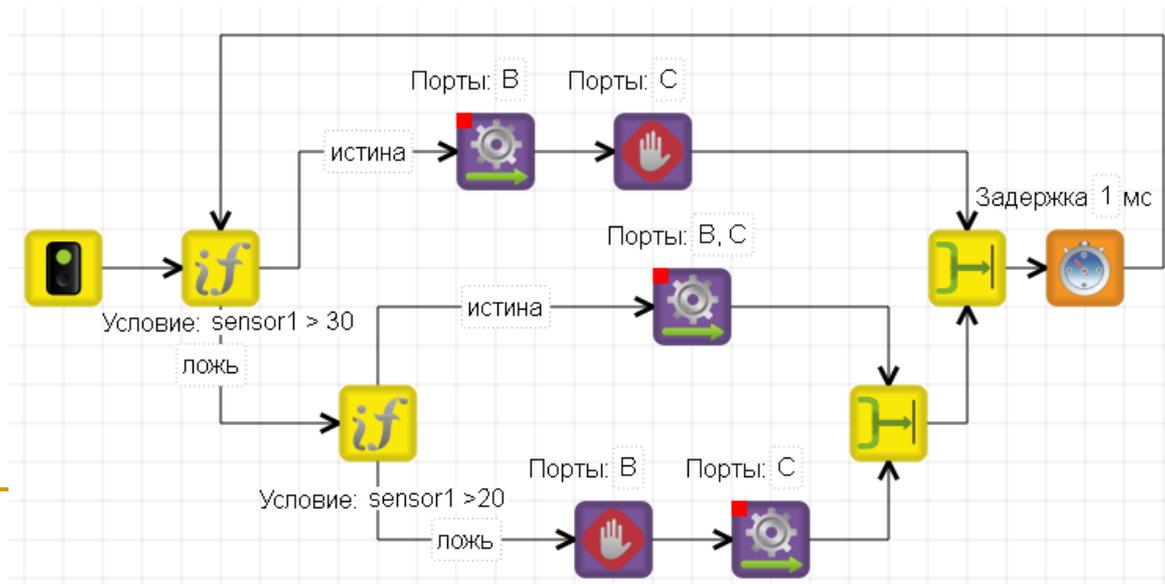
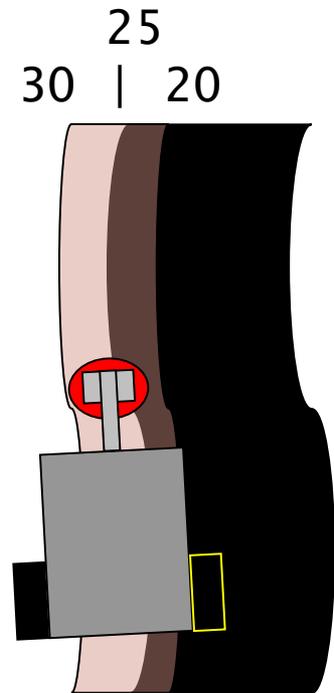
Вложенное ветвление

- Вложенное ветвление находится на одной из ветвей основного ветвления



Трехпозиционный релейный регулятор

- Рассмотрим окрестность вокруг значения серого ± 5 процентов
свет > 30 | $30 \geq \text{свет} > 20$ | свет ≤ 20
белый | серый | черный
- Если показания датчика попадают в окрестность, робот движется прямолинейно – новое действие



Пропорциональный регулятор

В задачах автоматического регулирования **управляющее воздействие u** обычно является функцией динамической ошибки – отклонения **e** регулируемой величины **x** от ее заданного значения **x_0** :

$$e = x_0 - x$$

Пропорциональный регулятор – это устройство, оказывающее управляющее воздействие на объект пропорционально его отклонению от заданного состояния.

$$u = k \cdot e$$

Здесь **k** – это коэффициент усиления регулятора.

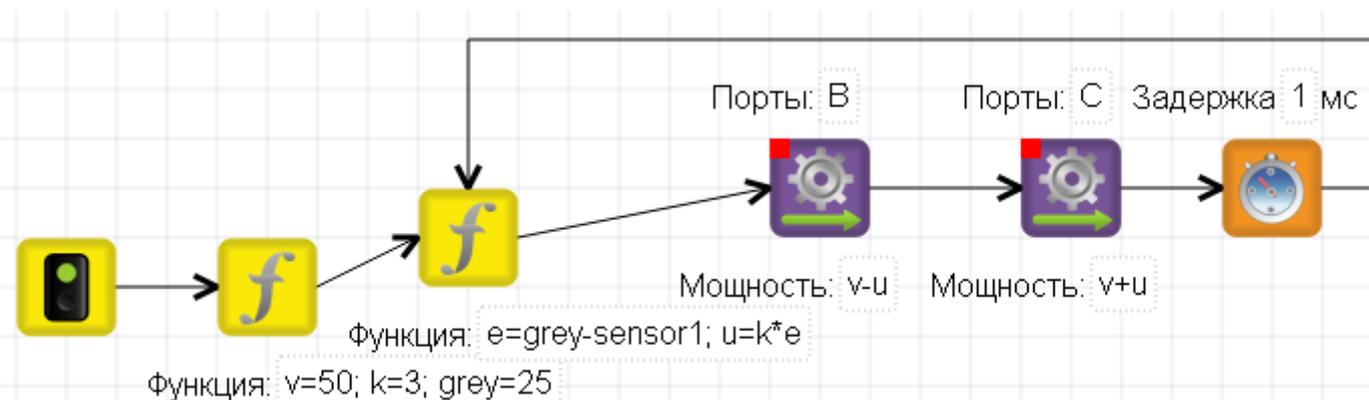


П р о п о р ц и о н а л ь н ы й р е г у л я т о р :

Д В И Ж Е Н И Е П О Л И Н И И

- Также как и в релейном регуляторе, необходимо определить среднее значение **grey** между черным и белым. Это будет то состояние датчика освещенности **sensor1**, к которому должна стремиться система.
- $e = \text{grey} - \text{sensor1}$

Видит человек



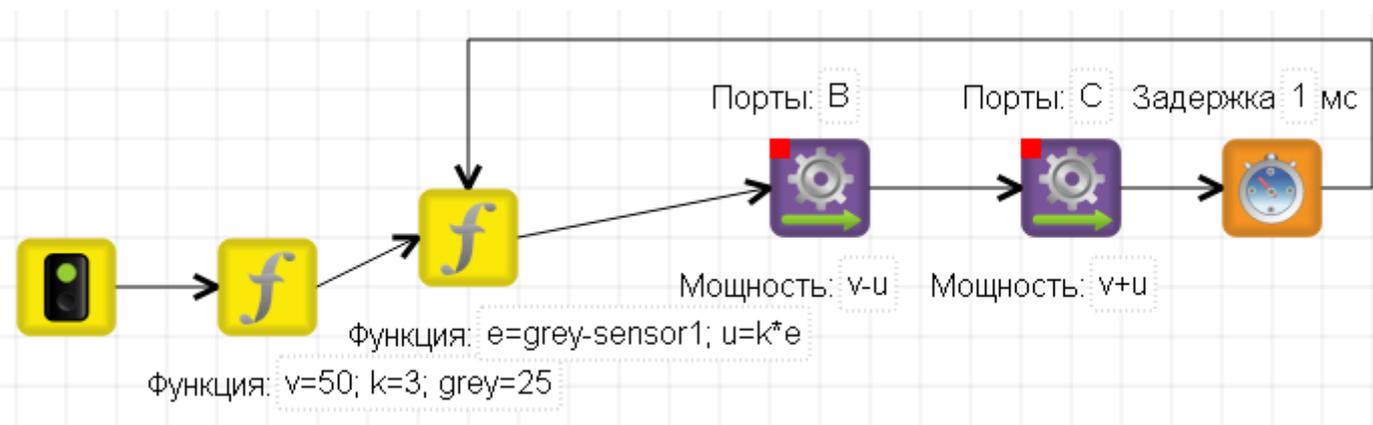
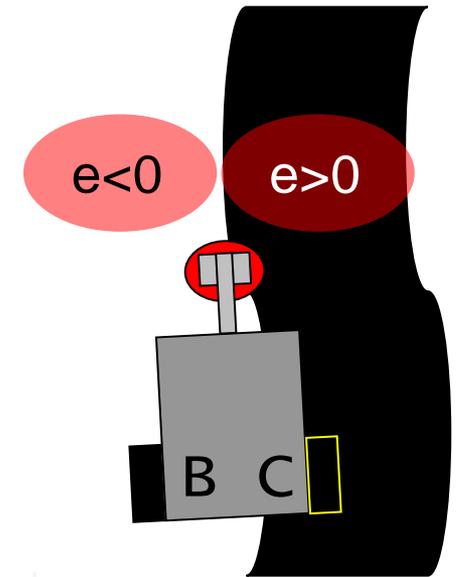
Видит робот



П р о п о р ц и о н а л ь н ы й р е г у л я т о р :

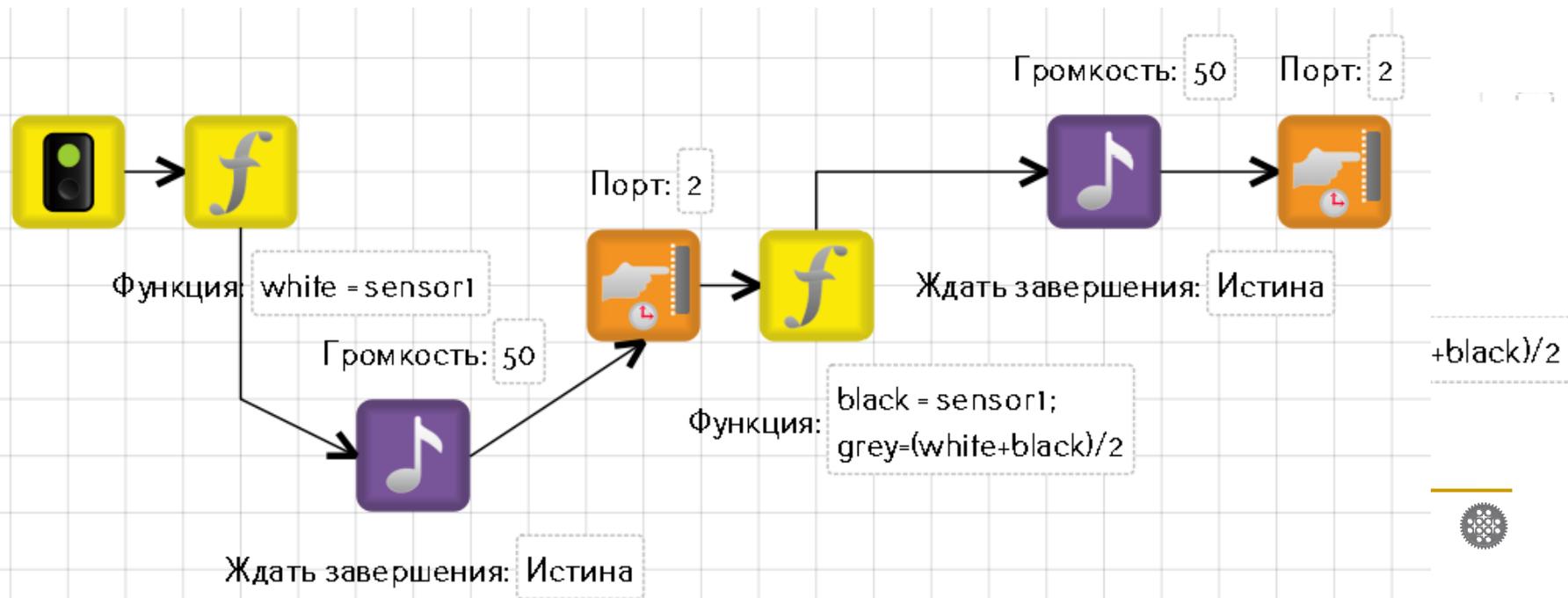
Д В И Ж Е Н И Е П О Л И Н И И

- Также как и в релейном регуляторе, необходимо определить среднее значение **grey** между черным и белым. Это будет то состояние датчика освещенности **sensor1**, к которому должна стремиться система.
- $e = grey - sensor1$



Калибровка датчика освещенности

- Полная калибровка с определением белого и черного
- Ожидание нажатия кнопки
- Добавить сигнал с задержкой



Вывод на экран результата калибровки

- Полная калибровка с определением белого и черного
- **Вывод результата на экран**
- Ожидание нажатия датчика касания
- Выведите значения **white**, **black** и **grey**

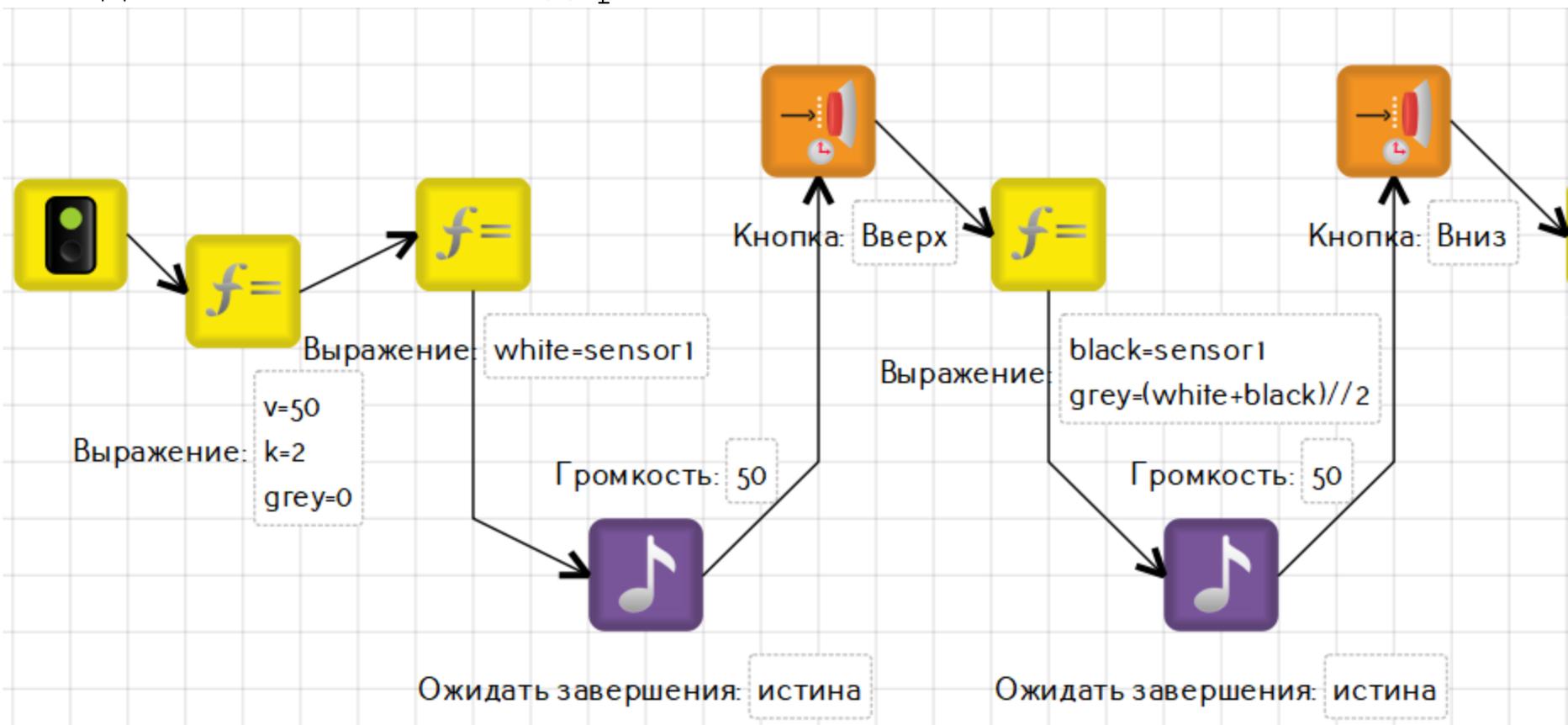
The image shows a graphical programming environment. On the left, a flowchart consists of three blocks: a mouse icon, a green block with the text 'ABC', and an orange block with a hand icon and 'Порт: 2'. Arrows connect them in sequence. Below the 'ABC' block, a tooltip shows 'X: 1', 'Y: 1', and 'Текст: grey'. On the right, a 'Редактор свойств' (Properties Editor) window is open, displaying a table of properties and their values.

| Свойство | Значение |
|-------------------|--|
| Вычислять | <input checked="" type="checkbox"/> истина |
| Текст | grey |
| Обновить картинку | <input checked="" type="checkbox"/> истина |
| X | 1 |
| Y | 1 |



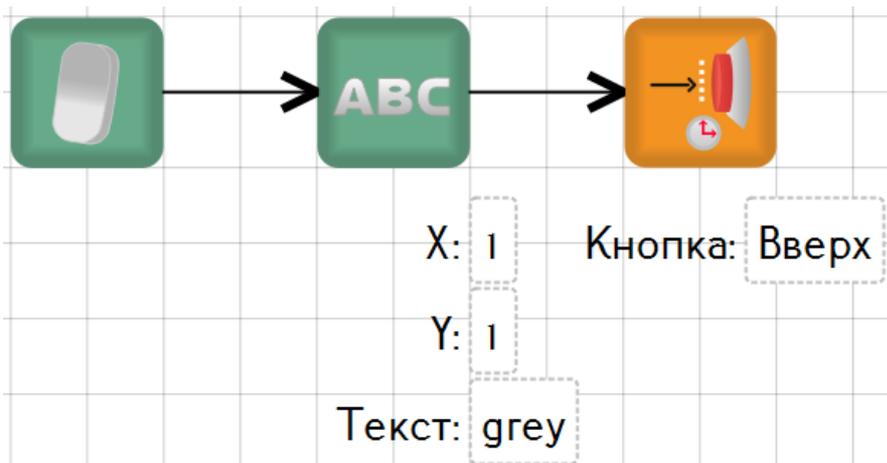
Калибровка с ожиданием кнопок EV3

- Полная калибровка с определением белого и черного
- Ожидание нажатия разных кнопок EV3
- Добавить сигнал с задержкой



Вывод на экран с ожиданием нажатия кнопки EV3

- Полная калибровка с определением белого и черного
- **Вывод результата на экран**
- Ожидание нажатия кнопки



| Свойство | Значение |
|-------------------|--|
| Вычислять | <input checked="" type="checkbox"/> истина |
| Текст | grey |
| Обновить картинку | <input checked="" type="checkbox"/> истина |
| X | 1 |
| Y | 1 |

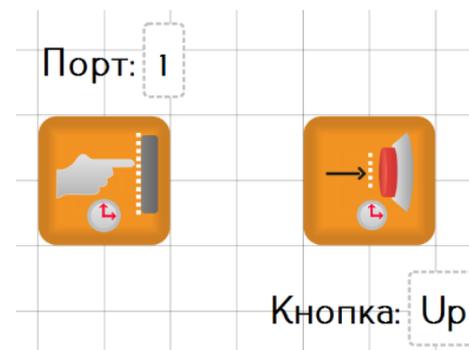
Редактор свойств Настройки сенсоров



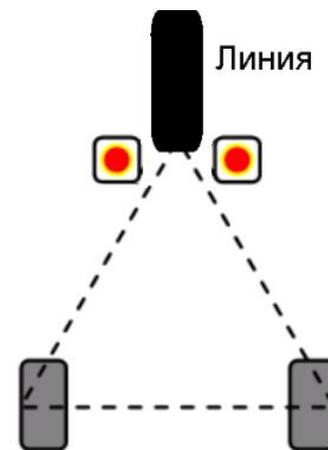
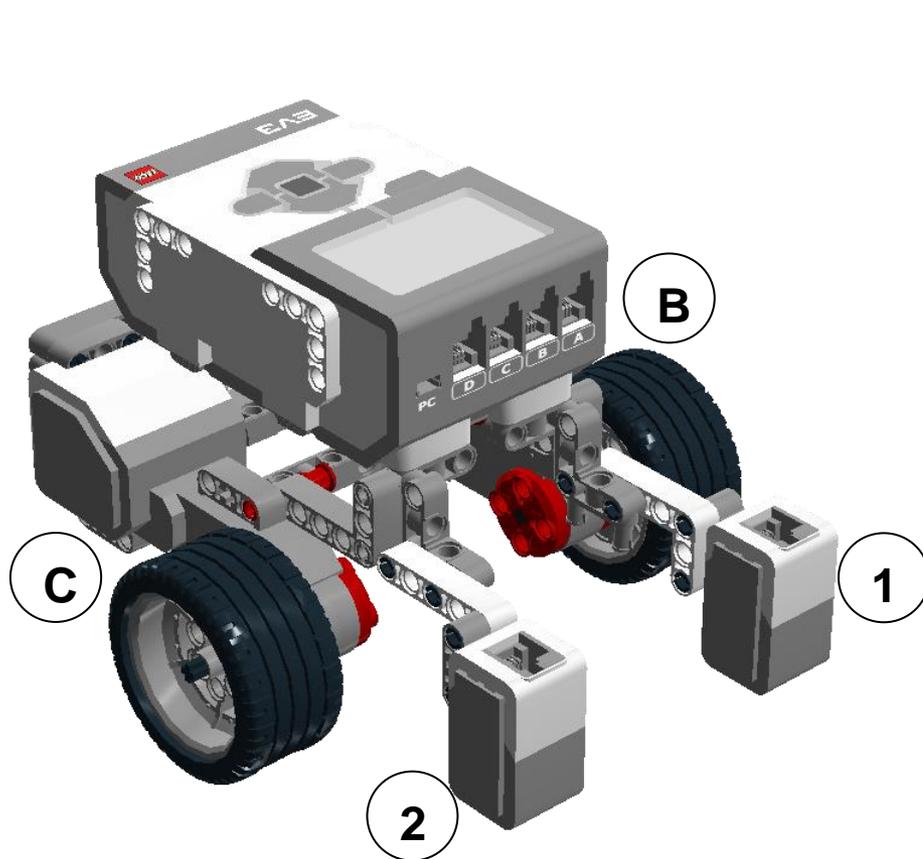
Краткая калибровка датчика

ОСВЕЩЕННОСТИ

- Краткая калибровка с понижением, датчик над белым полем
- $grey = sensor1 - (white - black) / 2$
- Значения **white** и **black** надо выяснить заранее

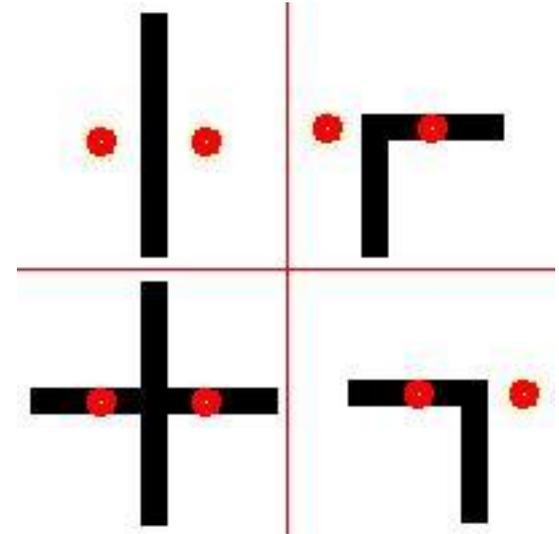


Движение по линии с двумя датчиками освещенности



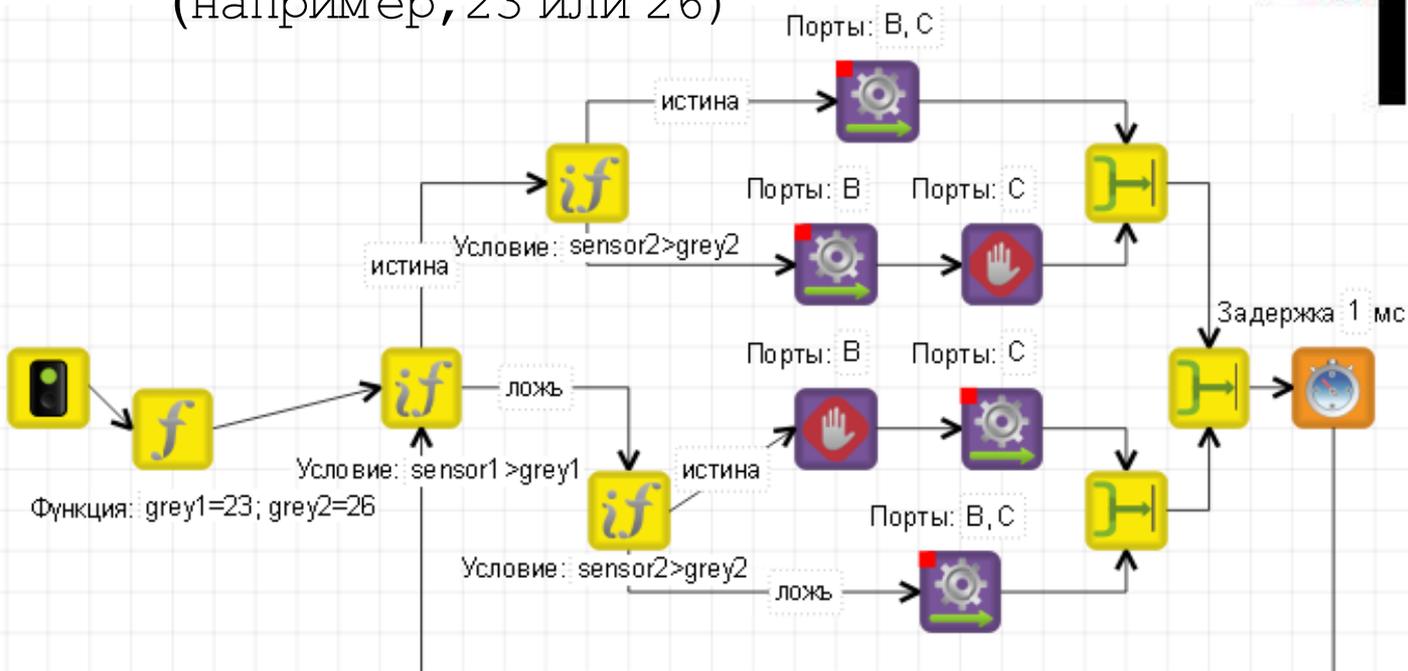
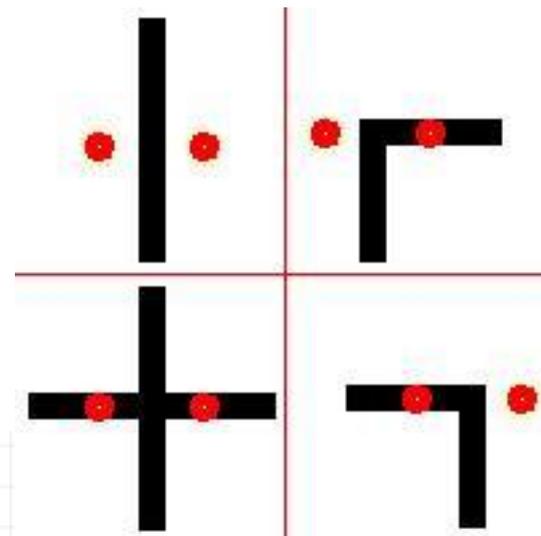
Релейный регулятор : движение с двумя датчиками освещенности

- Оба на белом – движение прямо,
- левый (**sensor1**) на черном, правый (**sensor2**) на белом – движение налево,
- левый на белом, правый на черном – движение направо,
- оба на черном – движение прямо.



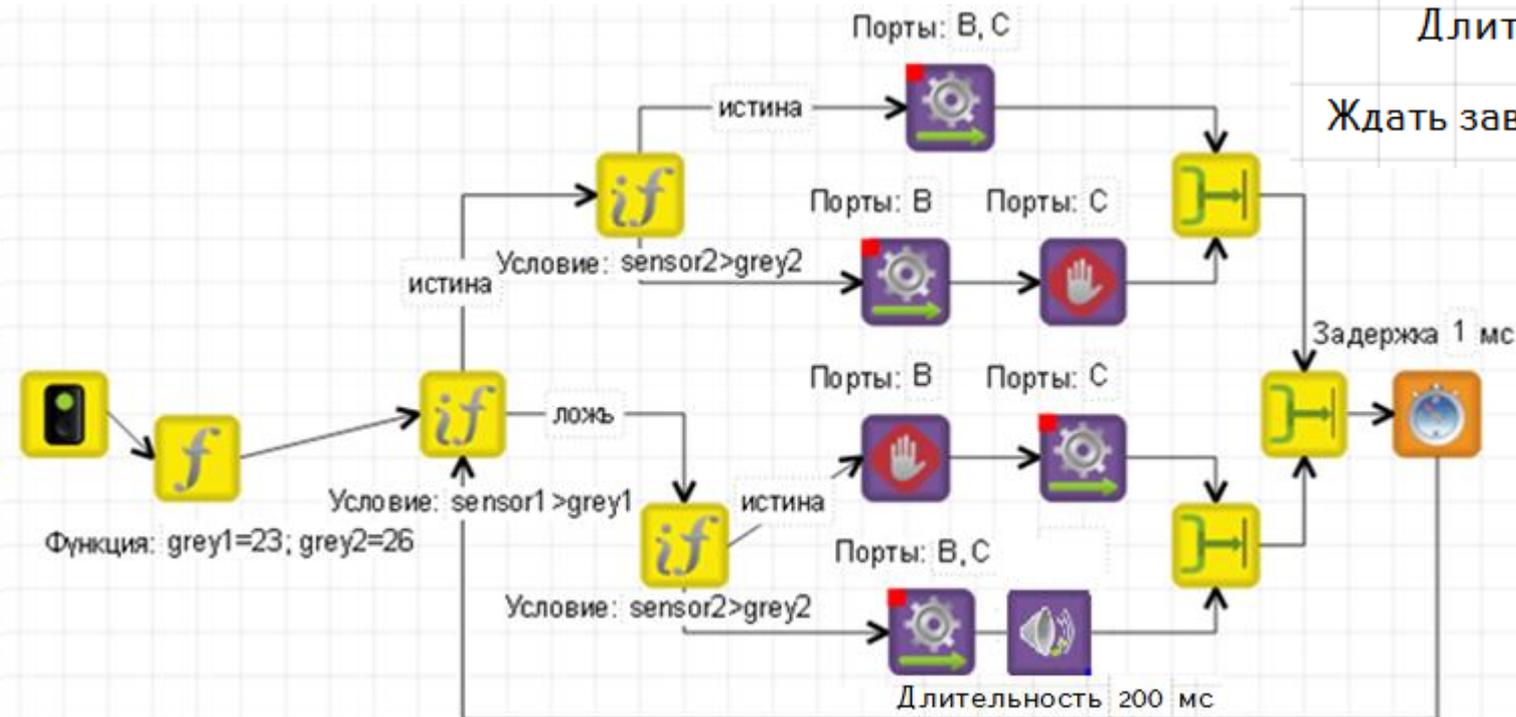
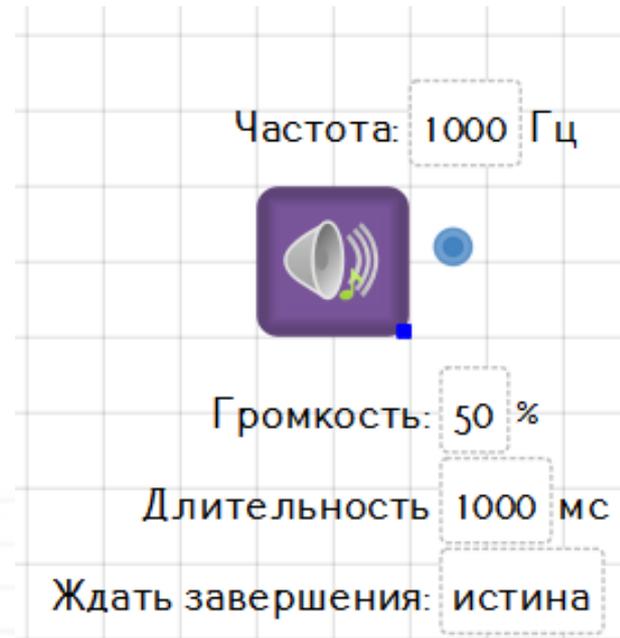
Релейный четырехпозиционный регулятор : вложенное ветвление

- На каждой ветви проверки первого датчика идет проверка второго датчика
- Алгоритм выполняется только по одной из ветвей
- Для каждого датчика требуется определить свое значение серого (например, 23 или 26)

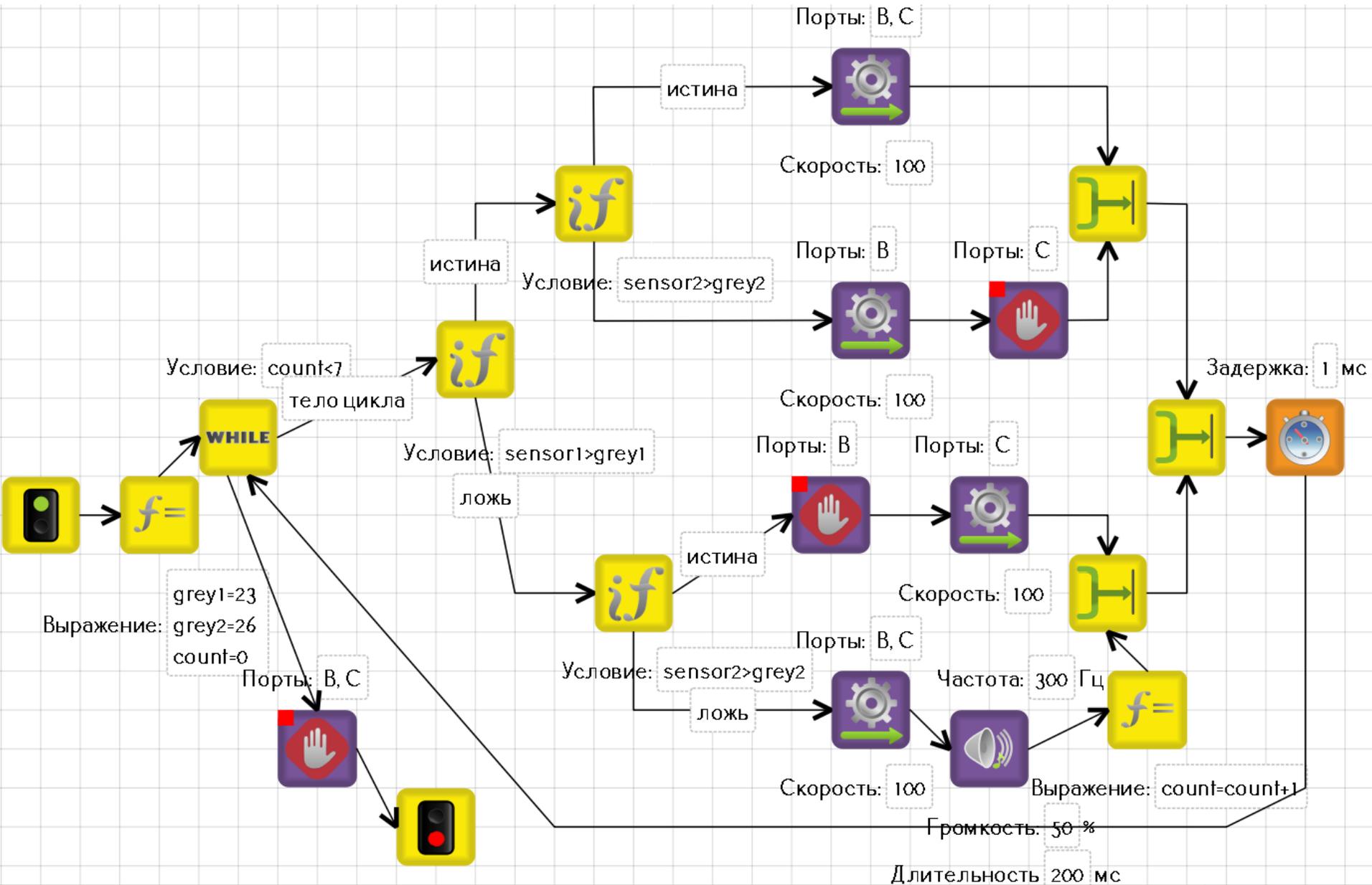


Реакция на перекрестки

- Издать звуковой сигнал на каждом перекрестке
- При звуковом сигнале без ожидания один перекресток фиксируется несколько раз
- При длительном ожидании робот может сойти с линии
- Ожидание не должно превышать 200 мс

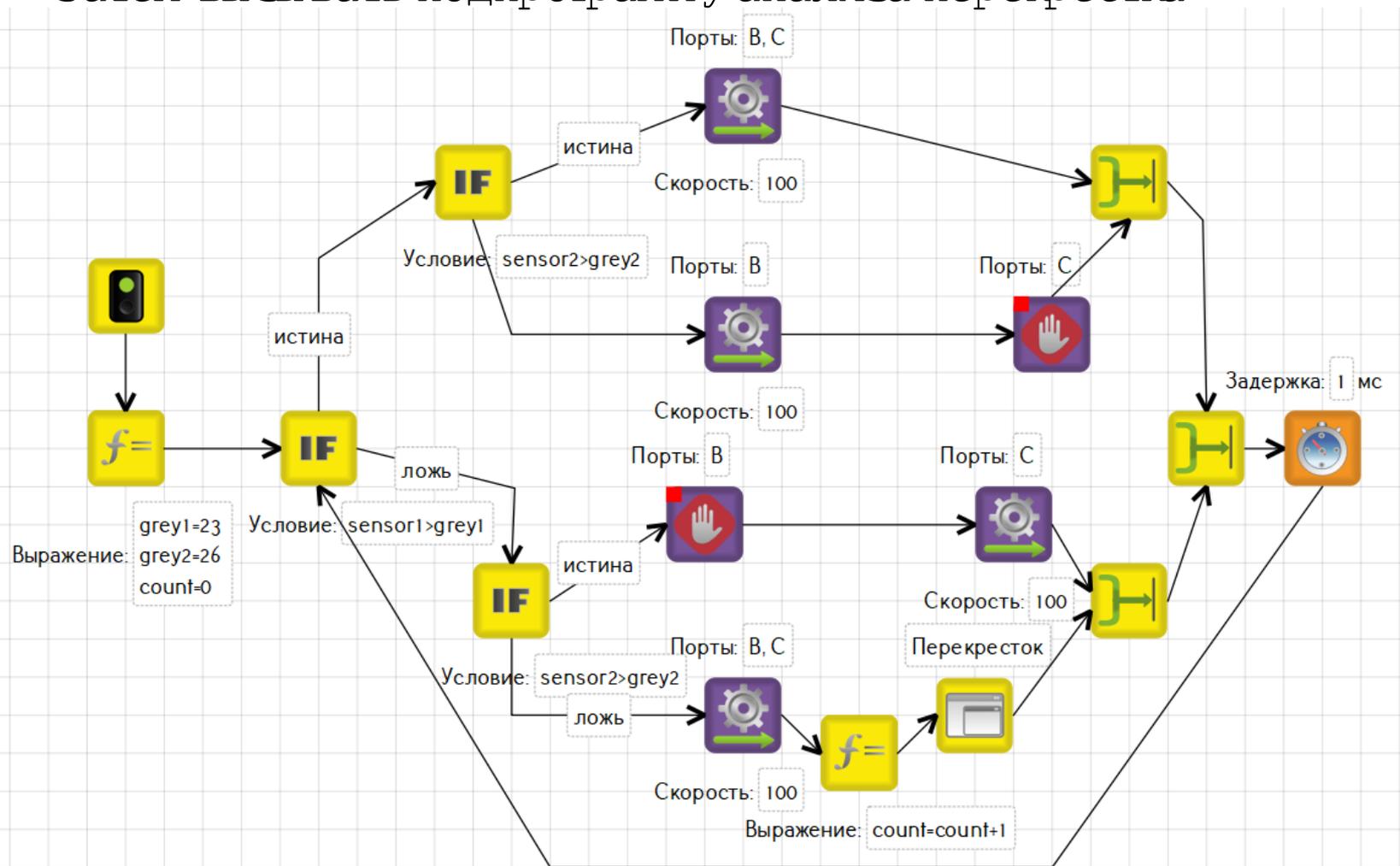


П подсчет перекрестков



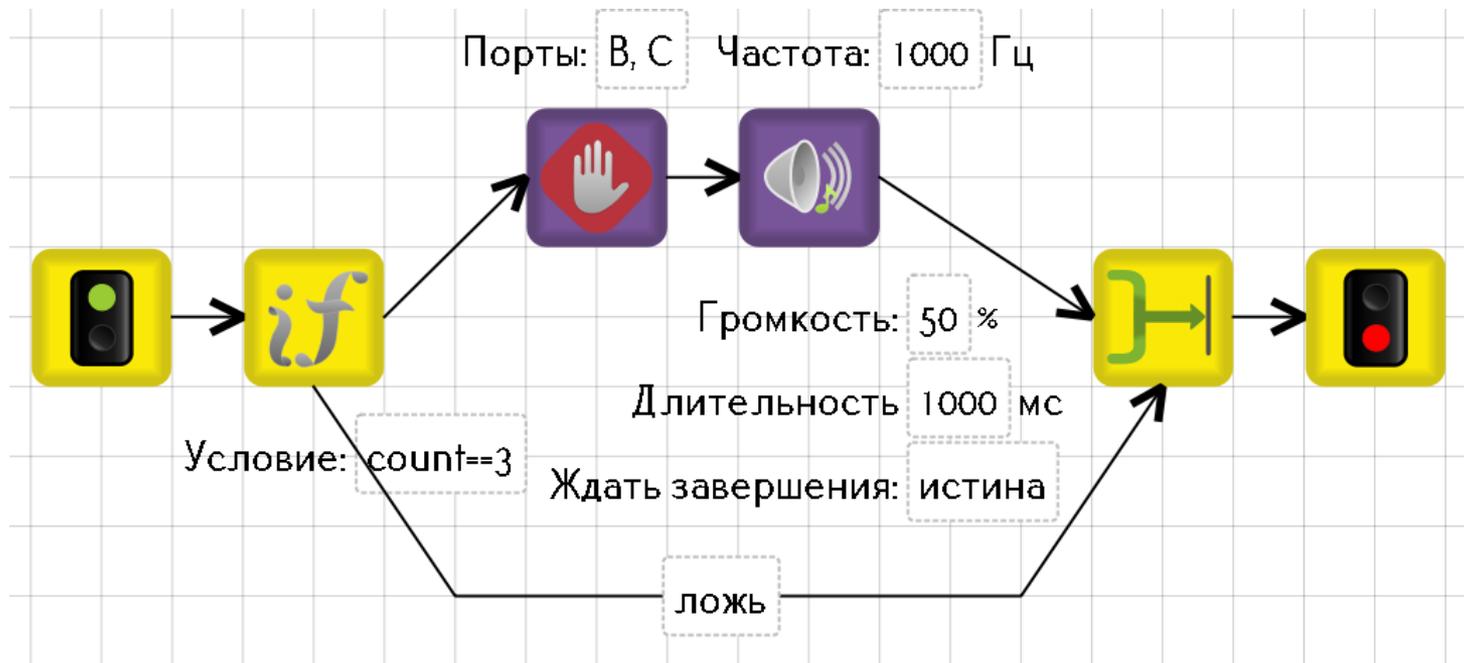
Действия на перекрестках

- На старте обнулить счетчик перекрестков $count=0$
- На каждом перекрестке увеличивать счетчик на 1
- Затем вызывать подпрограмму анализа перекрестка



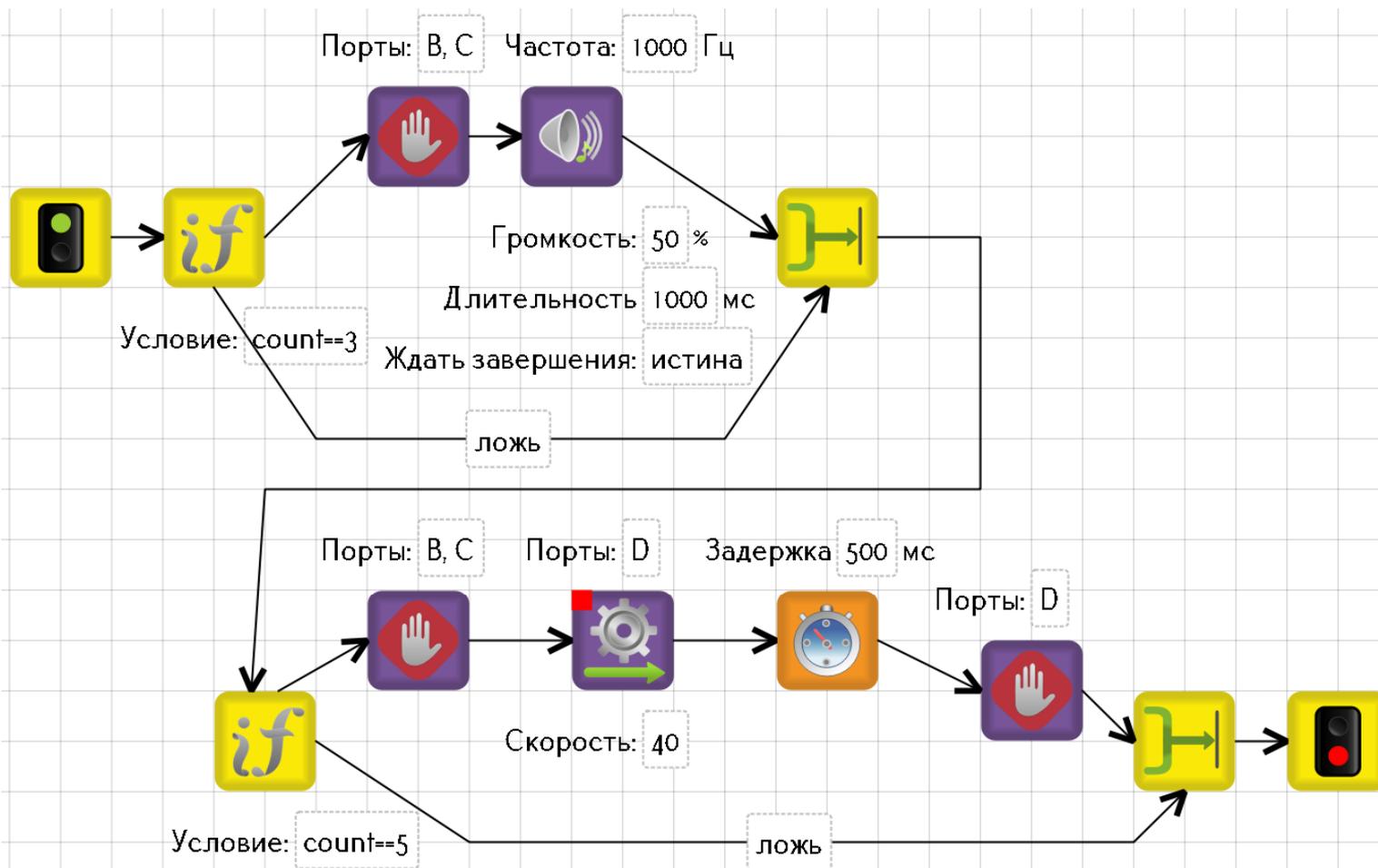
Действия на перекрестках

- На заданном перекрестке `count==3` остановиться и издать звуковой сигнал
- Для наглядности создать подпрограмму и вызвать ее после увеличения `count=count+1`
- Для действия на другом перекрестке надо добавить в эту программу еще одну проверку условия



Действия на перекрестках

- На 5 перекрестке сбить объект (после сигнала на 3 перекрестке)



Вывод номера перекрестка на экран

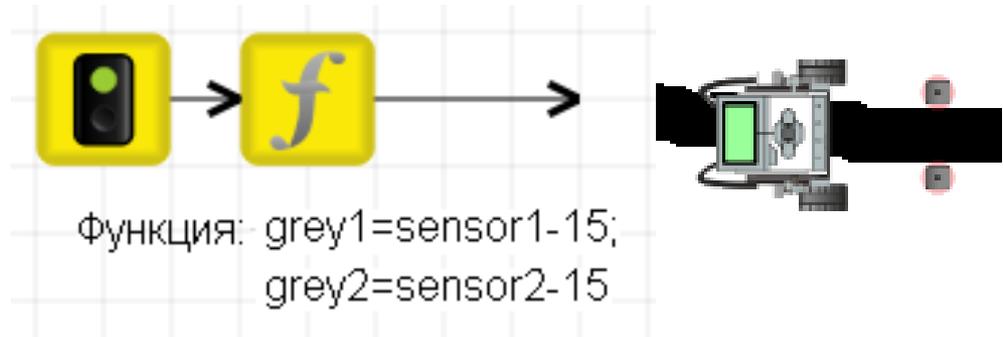
- В начало подпрограммы добавить вывод номера перекрестка
- В редакторе свойств блока **ABC** для пункта «Вычислять» поставить галочку «истина»

| Свойство | Значение |
|-------------------|--|
| Вычислять | <input checked="" type="checkbox"/> истина |
| Текст | count |
| Обновить картинку | <input checked="" type="checkbox"/> истина |
| X | 1 |
| Y | 1 |



Калибровка с понижением для двух датчиков

- Краткая калибровка



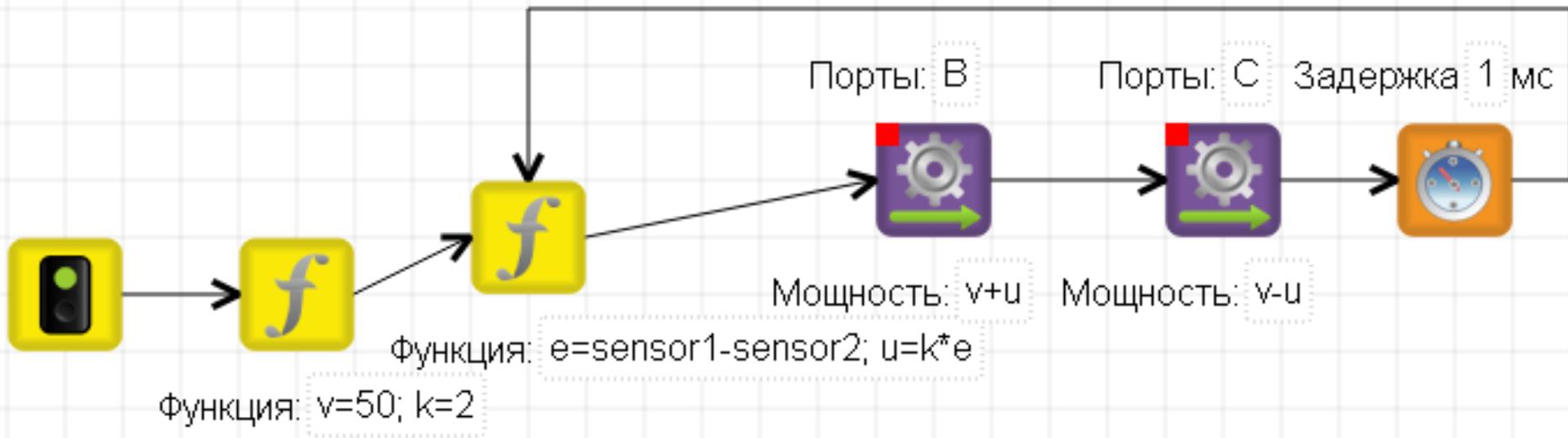
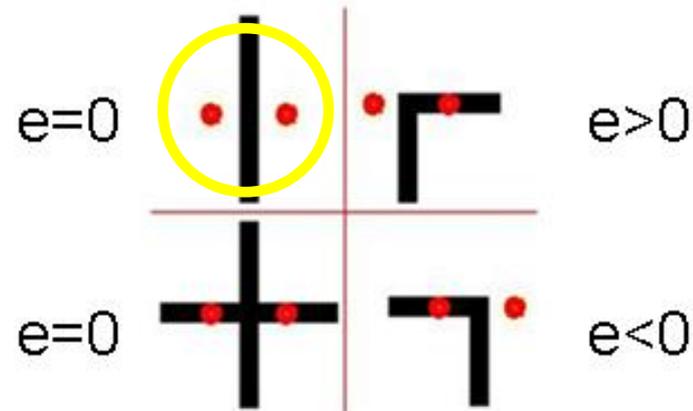
- Откуда взято вычитаемое число?
- Оно определено заранее как половина разности крайних показаний датчика
- $grey1 = sensor1 - (white1 - black1) / 2$
- $grey2 = sensor2 - (white2 - black2) / 2$
- Для точной калибровки желательно чистое одноцветное поле вне линии
- Для надежности следует сделать паузу перед началом движения
- Ожидание нажатия кнопки



Пропорциональный регулятор:

Движение по линии с двумя датчиками

- Ошибка рассматривается как разность показаний датчиков
- Нет зависимости от показаний на сером
- $e = \text{sensor1} - \text{sensor2}$



Пропорциональный регулятор устранение статической ошибки

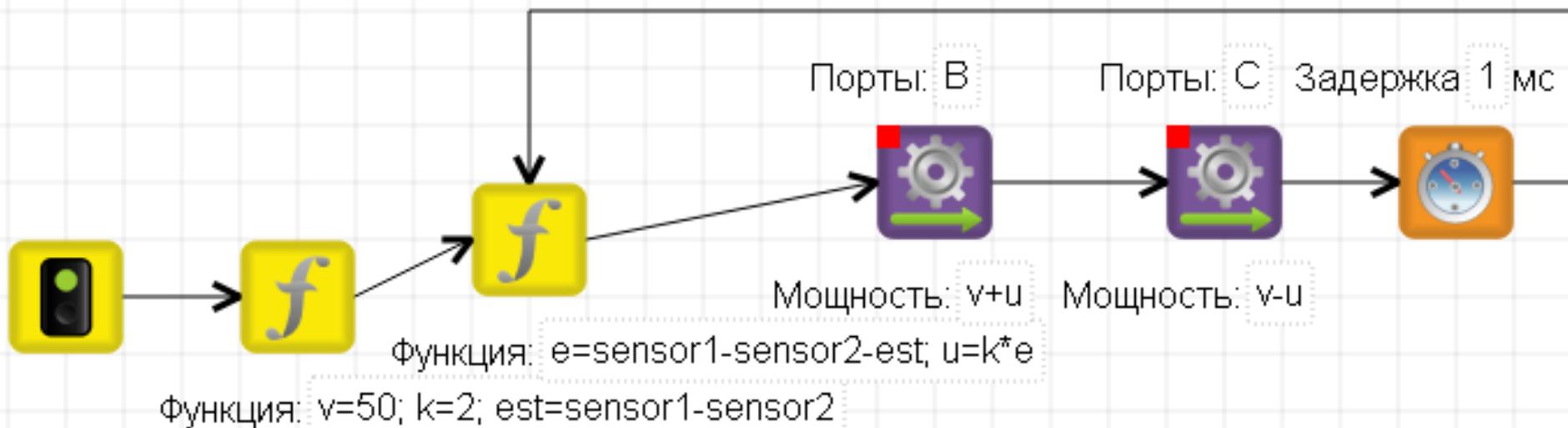
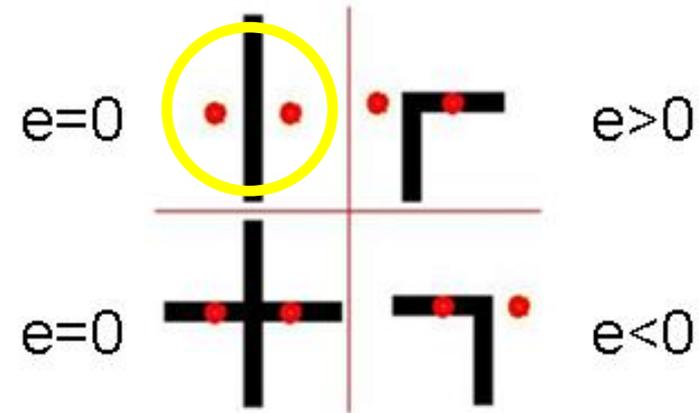
- Датчики могут быть изготовлены с отклонениями
- Разность показаний на однотонной поверхности - это **статическая ошибка**

До цикла

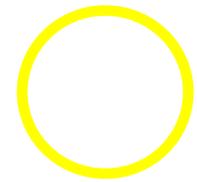
- $est = sensor1 - sensor2$

В цикле статическая ошибка устраняется:

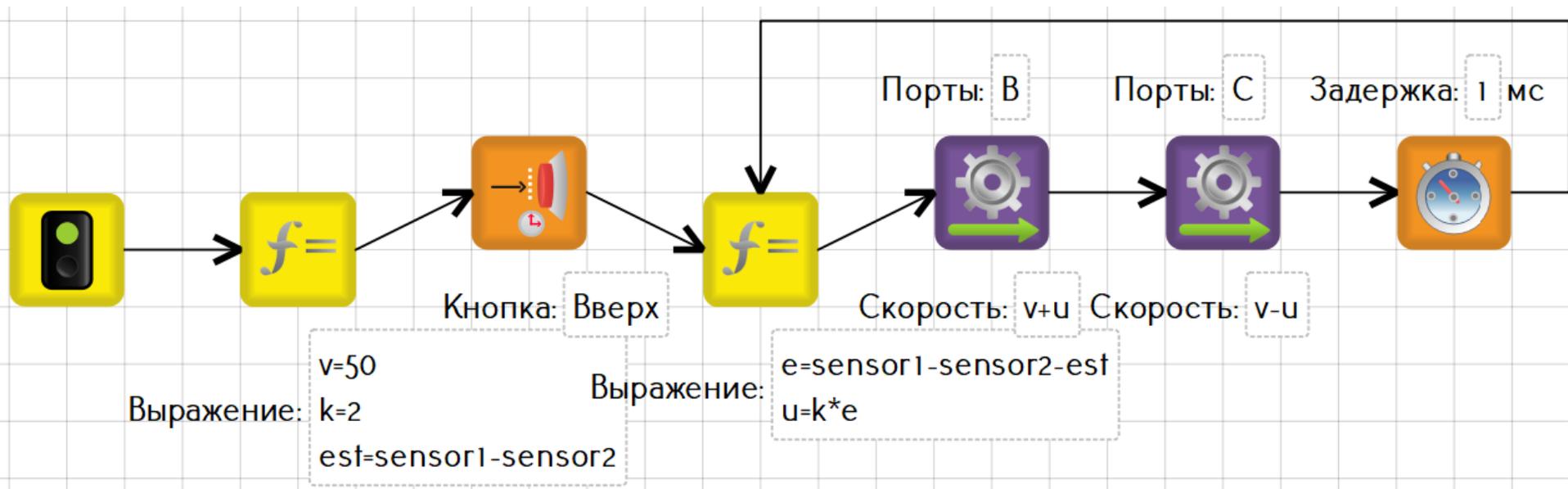
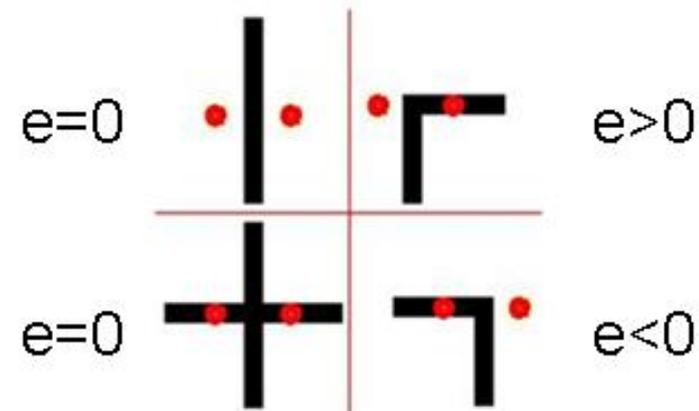
- $e = sensor1 - sensor2 - est$



Пропорциональный регулятор устранение статической ошибки и бки

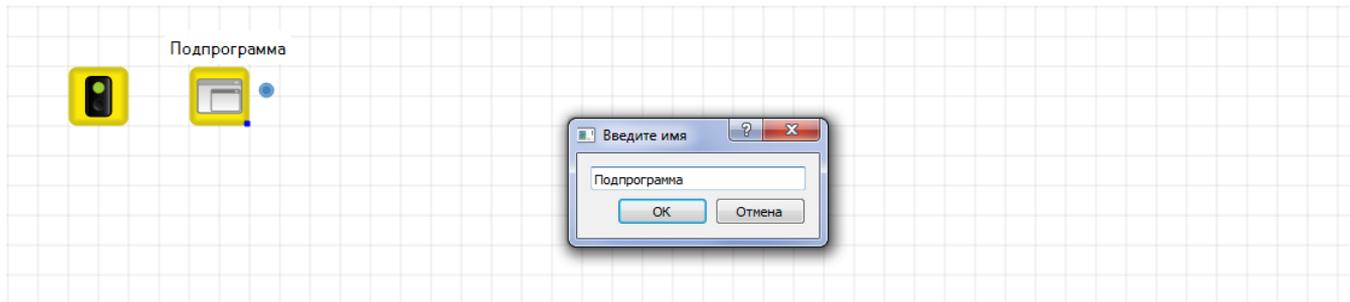


- Считывание статической ошибки и бки на белом поле в стороне от линии
- Ожидание нажатия кнопки EV3 перед стартом



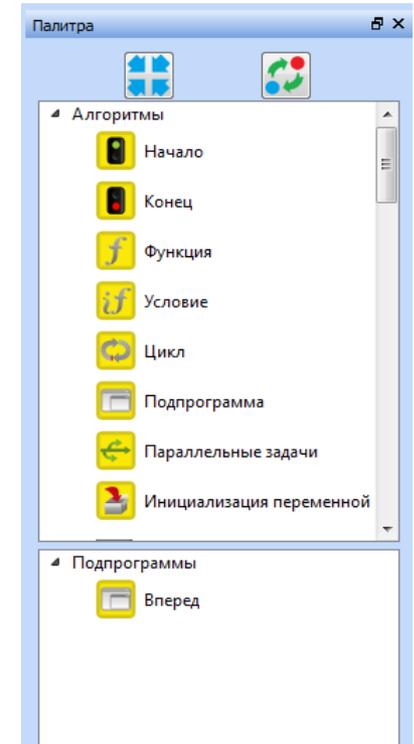
Подпрограммы

Вытащите первый блок «Подпрограмма» на сцену



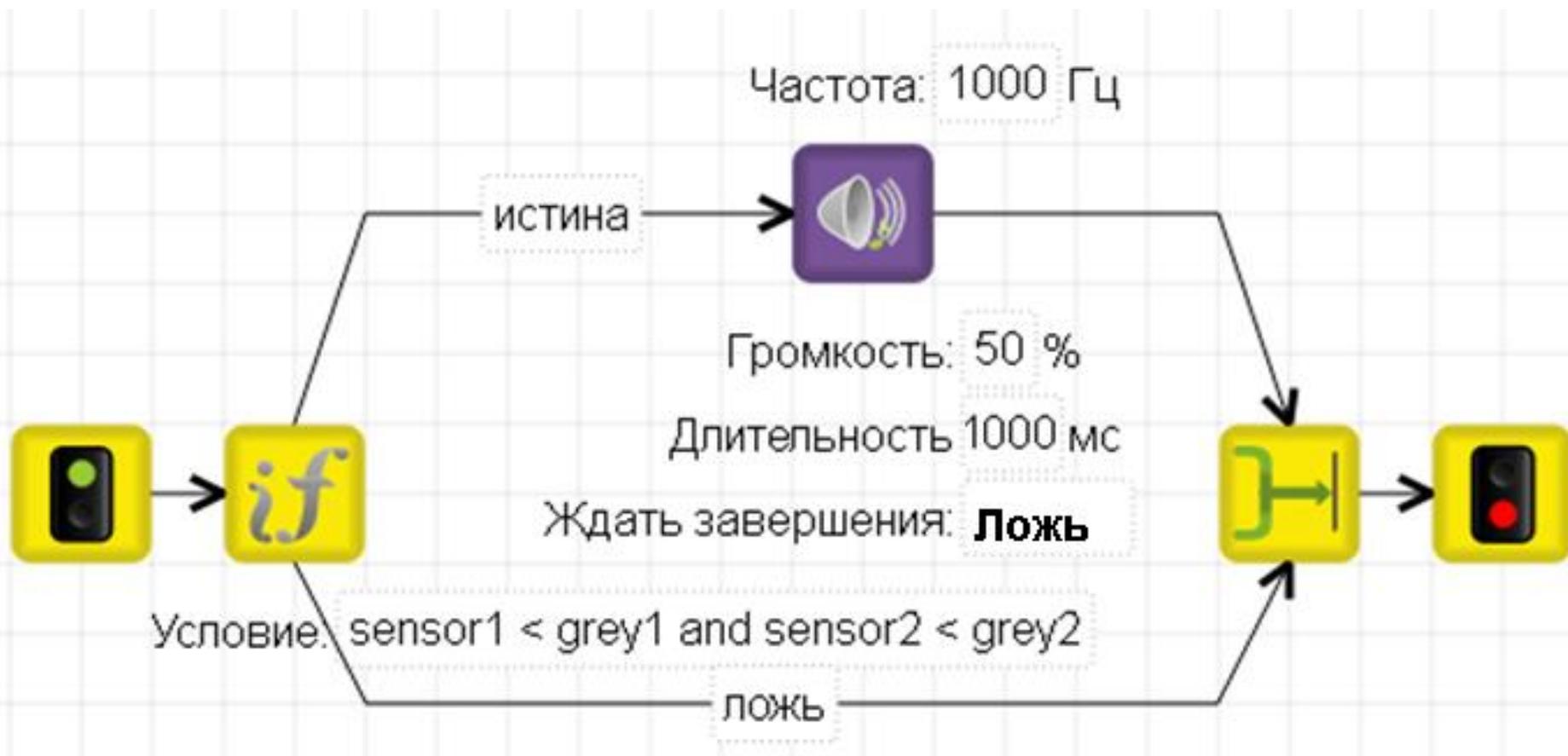
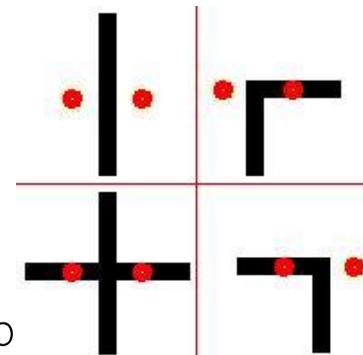
Назовите её. Блок подпрограммы появится в палитре

Двойным щелчком по подпрограмме перейдем к диаграмме её алгоритма



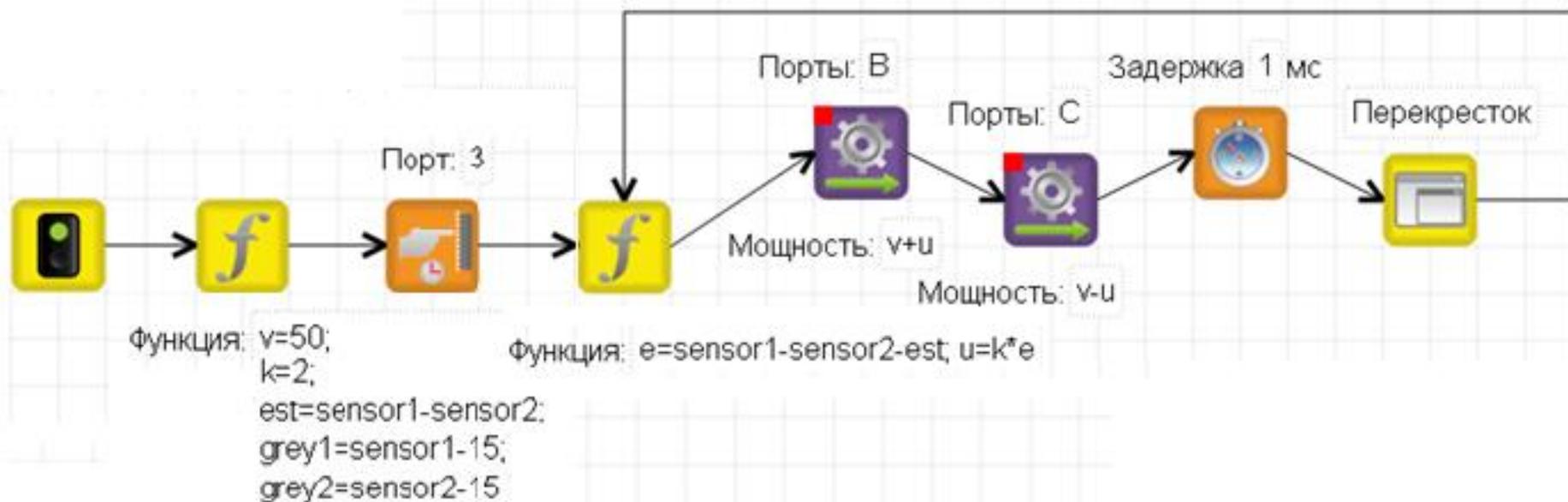
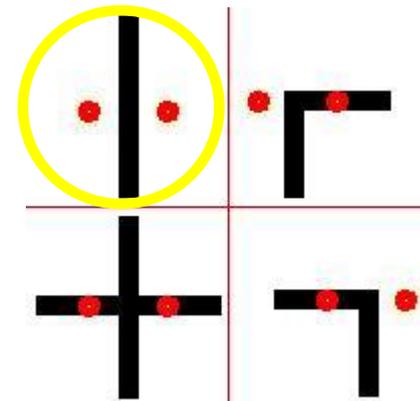
Реакция на перекрестки

- Издать звуковой сигнал на перекрестке
- Из-за задержки во время сигнала робот может сбиться с линии, поскольку он долго едет вслепую



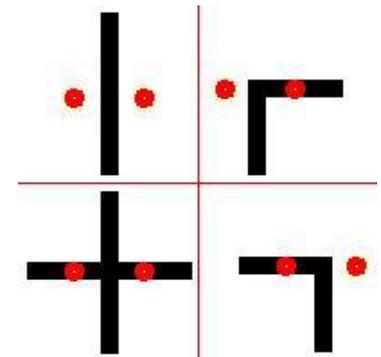
Пропорциональный регулятор: калибровка с ожиданием

- Для точной калибровки желательно чистое одноцветное поле вне линии
- Для надежности следует сделать паузу перед началом движения
- Ожидание нажатия кнопки
- Необходимо объявить все переменные в основной программе



Реакция на перекрестки

- Издать звуковой сигнал на перекрестке
- Из-за задержки во время сигнала робот может сбиться с линии, поскольку он долго едет вслепую
- Нужен сигнал без ожидания завершения



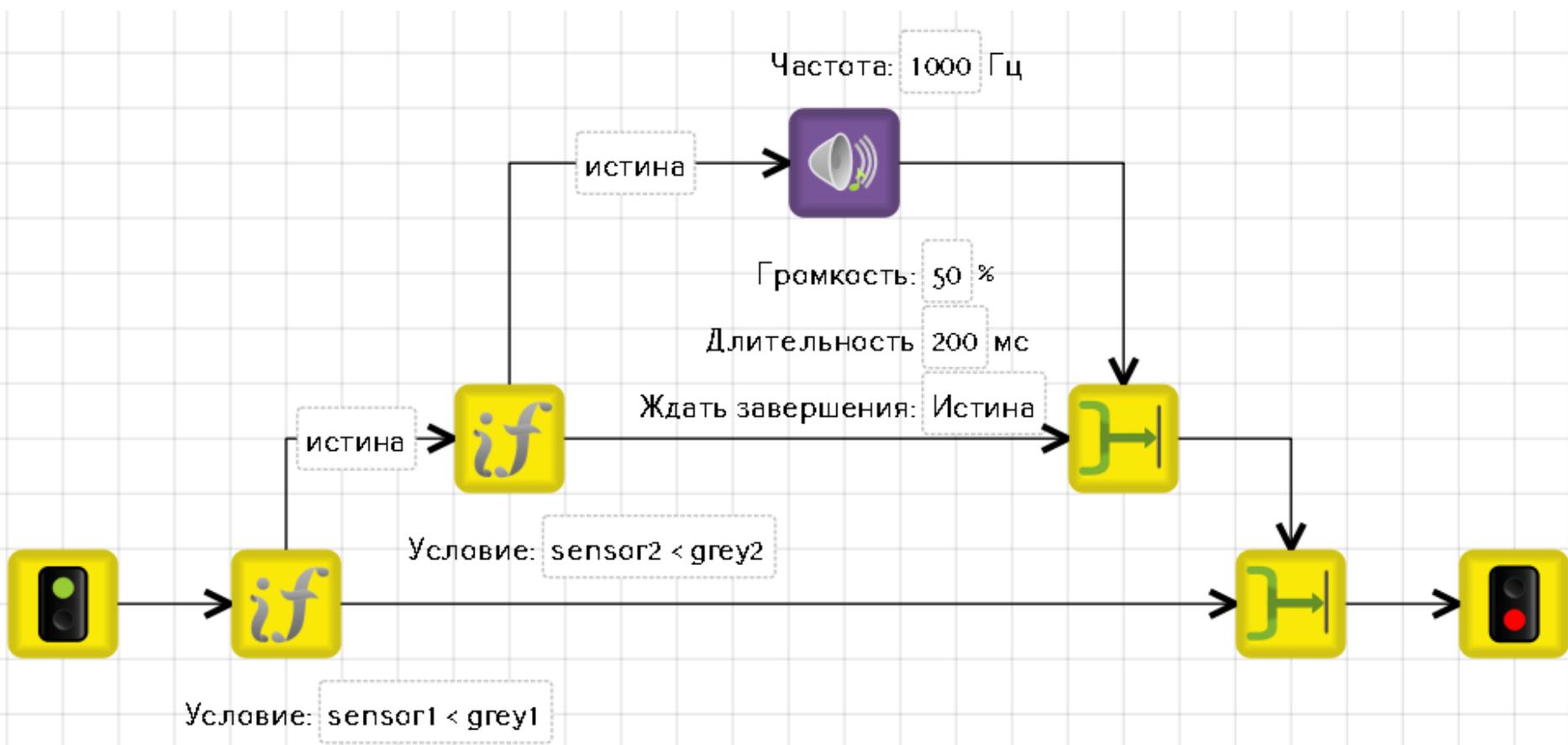
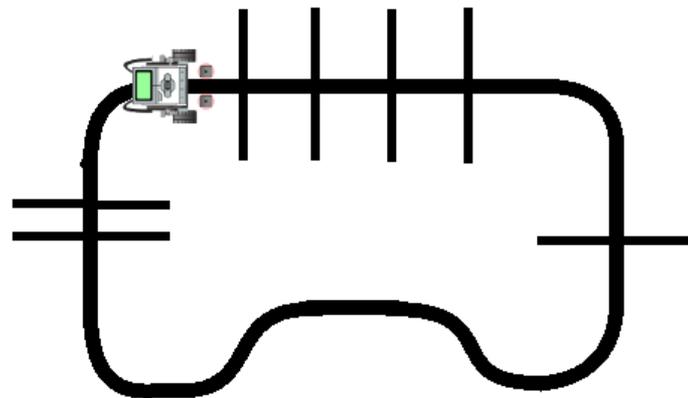
Реакция на перекрестки

- Сигнал без ожидания «пиликает» постоянно...
- Почему?
- Скорость робота – примерно 30 см /с
- Ширина линии перекрестка – 3 см
- На проезд линии робот тратит примерно 100 мс:
 - $3 \text{ см} / 30 \text{ см /с} = 0.1 \text{ с}$
- Значит, он успевает получить команду на сигнал около 100 раз
- Установим время движения «вслепую» 200 мс – с запасом



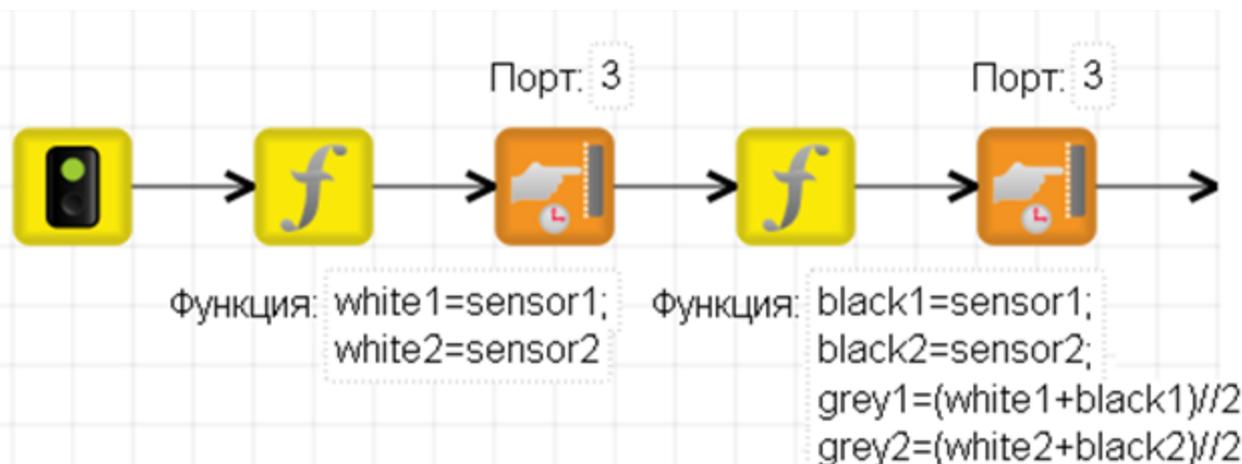
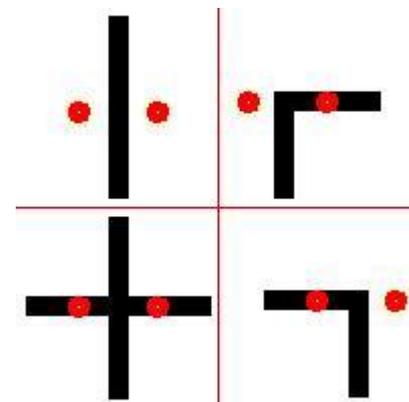
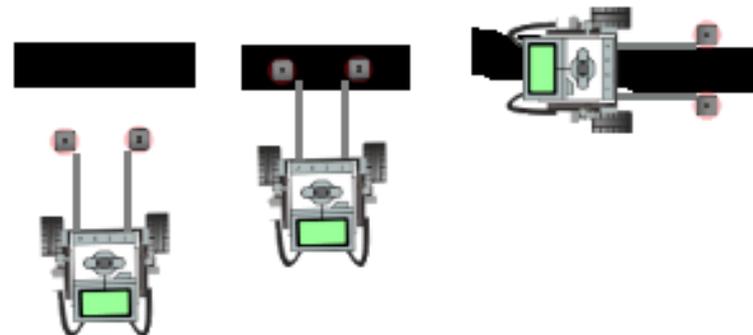
Реакция на перекрестки

- Вложенное ветвление – еще один способ проверки перекрестка



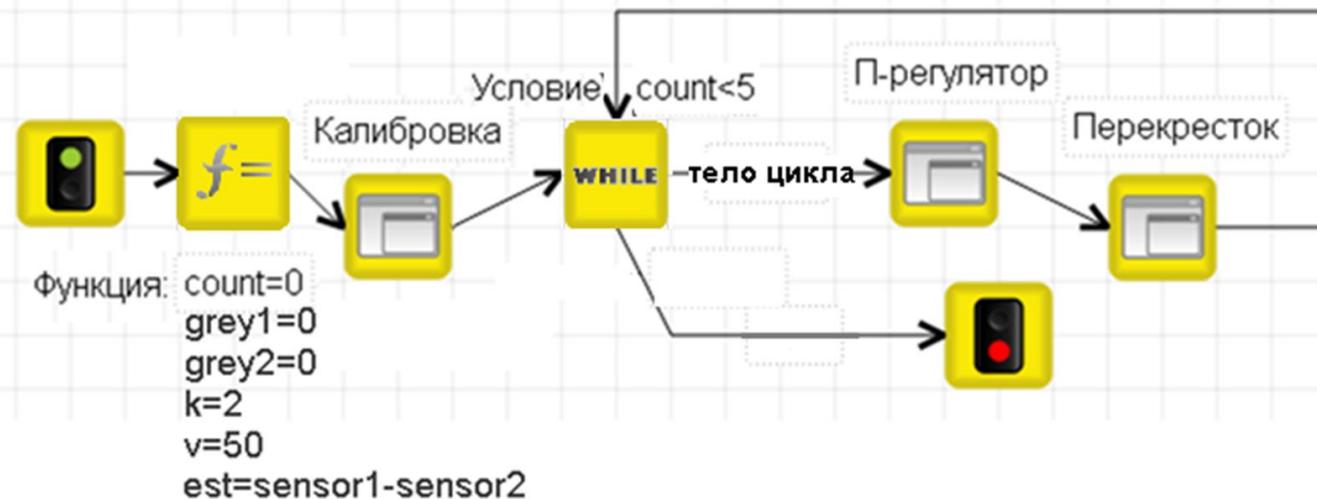
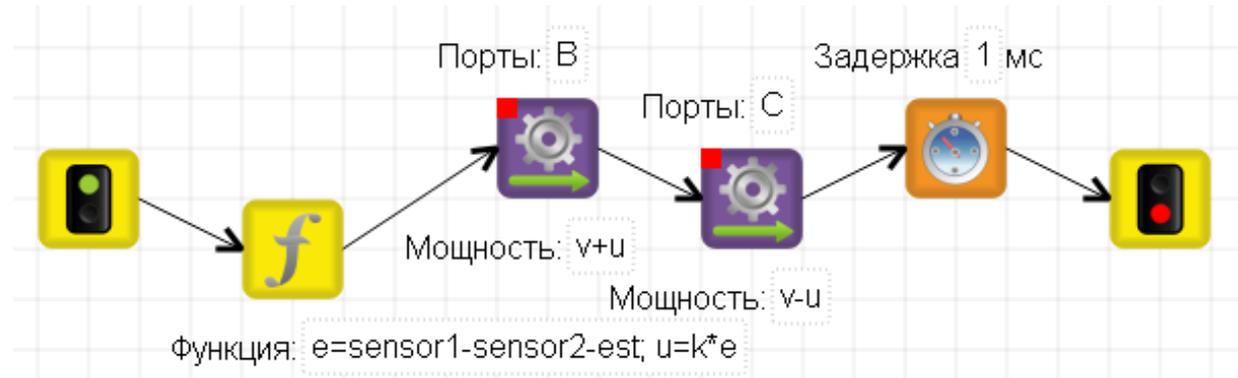
Полная калибровка для двух датчиков

- Для точной калибровки отдельно считывается черное, отдельно белое поле
- Ожидание нажатия кнопки
- Можно перенести в подпрограмму
- Потребуется объявить ключевые переменные в основной программе
 - `grey1=0`
 - `grey2=0`



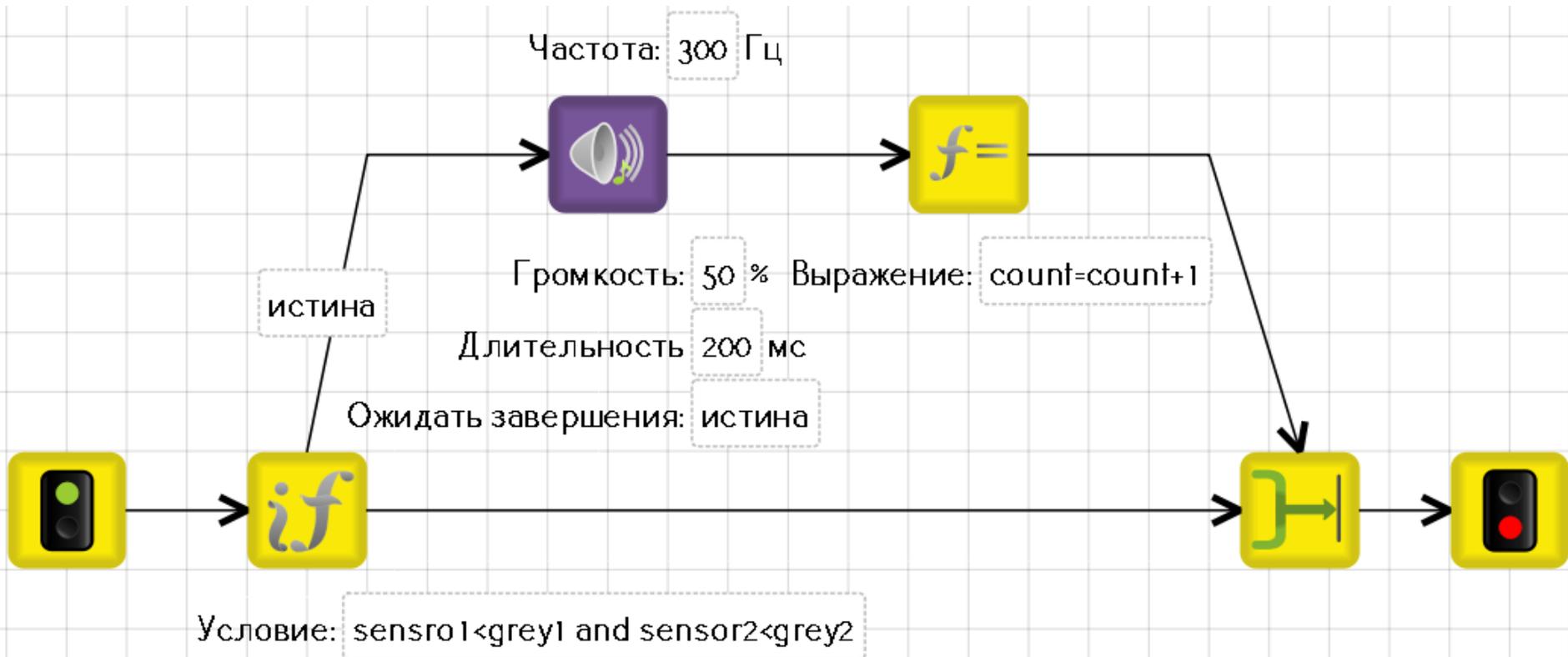
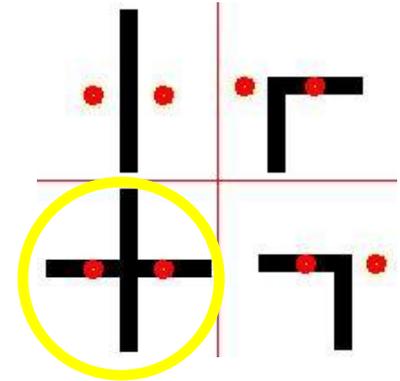
П одсчет перекрестков

- П-регулятор поместить в отдельную подпрограмму
- Обнуляем счетчик **count** и задаем все переменные до программы
- Повторяем цикл, пока значение счетчика меньше 5
- На каждом перекрестке добавляем в счетчик 1
 $count = count + 1$



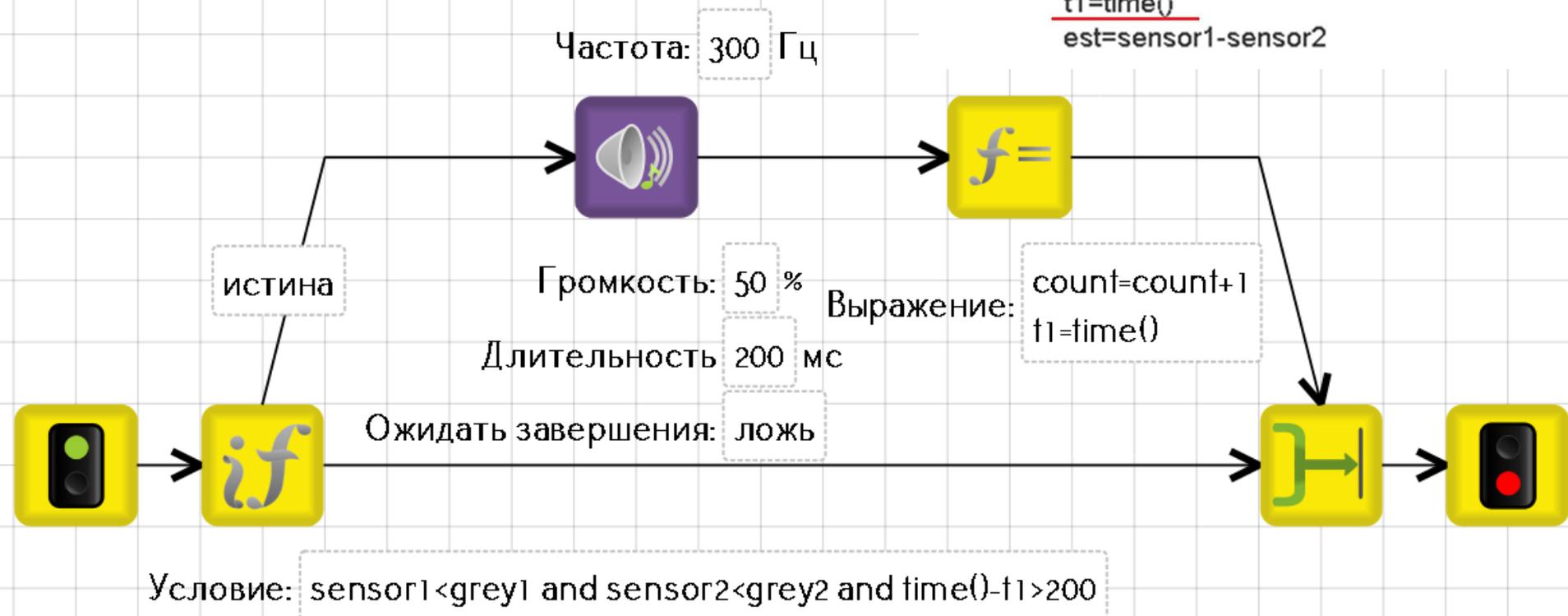
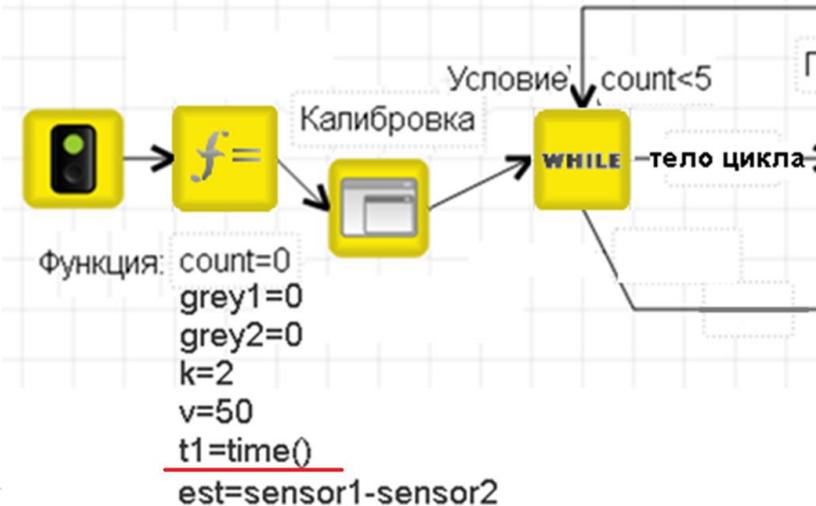
Увеличение счетчика

- На каждом перекрестке добавляем в счетчик 1:
 $\text{count} = \text{count} + 1$



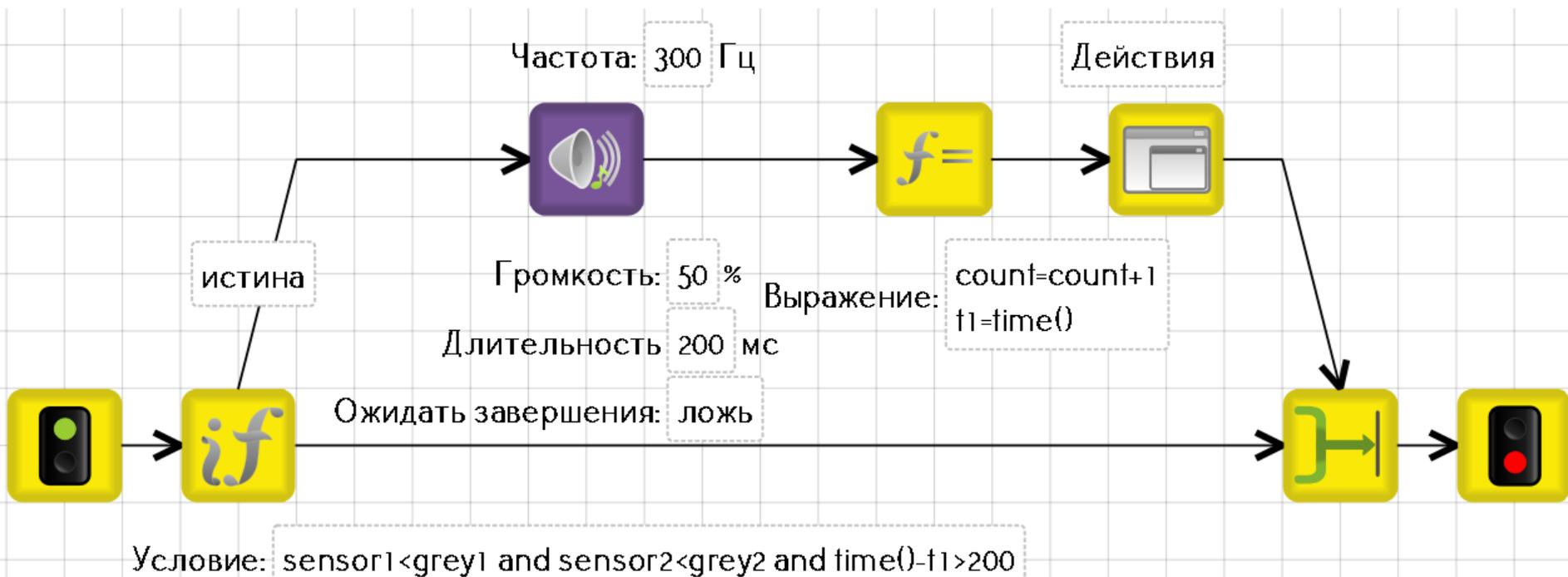
Усовершенствованный проезд перекрестка

- В начале запоминаем время
- На каждом перекрестке обновляем время
- Засчитываем перекресток только через 200 мс



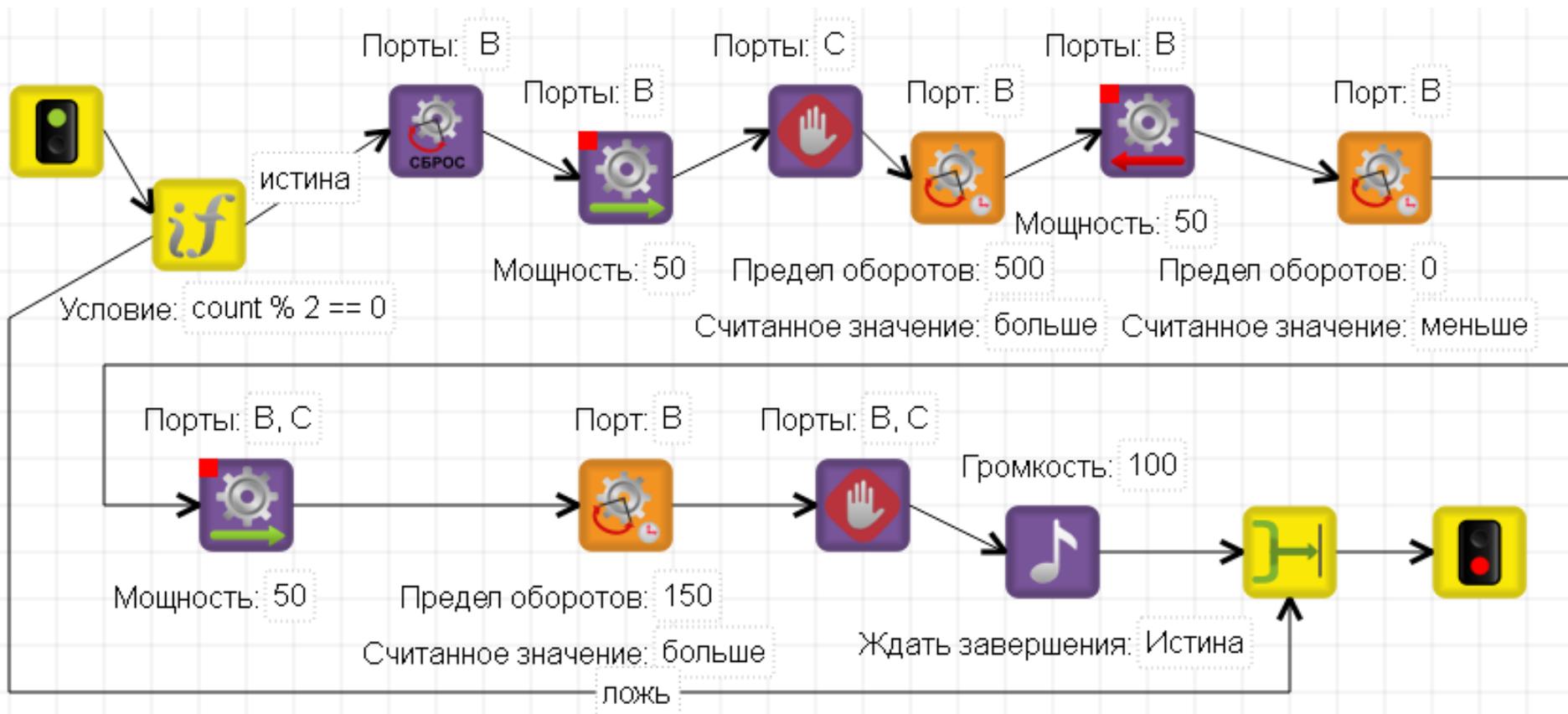
Действия на перекрестках

- Подпрограмма «Действия» добавляется сразу после увеличения счетчика



Реакция на объект

- Подпрограмма «Действия» сдвигает банки на каждом втором перекрестке
- Условие четности перекрестка :
- $\text{count} \% 2 == 0$



Благодарю за внимание!

Сергей Александрович Филиппов
Президентский физико-математический лицей № 239
Санкт-Петербург
safilippov@gmail.com

