

## Лабораторная работа №4

Тема. Программирование функций управления мобильным роботом для нахождения им заданных цветов в пространстве в среде Robotino View.

Цель: изучить новые функциональные блоки и элементы среды Robotino View. Составить программу для автоматического перемещения мобильного робота по заданным цветам.

Оборудование: мобильный робот, рабочий компьютер с установленными программами Robotino View и Robotino SIM.

### План работ

- 1) Краткая информация о блоках графического управления;
- 2) Описание работы программы для автоматического мобильного робота по заданным цветам;
- 3) Проверка работоспособности составленной программы реального мобильного робота или в среде Robotino SIM.

1. В данной лабораторной работе необходимо создать программу, которая позволит роботу поочередно искать и перемещаться по заданным цветам. Для этого нам необходимы следующие графические блоки:



(Robotino/Image system/Camera) – блок, подключающий камеру робота;



(Function block library/Image processing/Segmenter) – блок, присваивающий выбранным цветам порядковый номер;



(Function block library/Image processing/Segment Extractor) – блок поиска текущего цвета сегмента и получения данных о координатах его центра;



(Function block library/Image processing/Image Information) – блок выдачи ширины и высоты принимаемого изображения с камеры робота.

2. На рисунке 1 представлена структура выполнения главной программы с условиями перехода на следующие шаги.

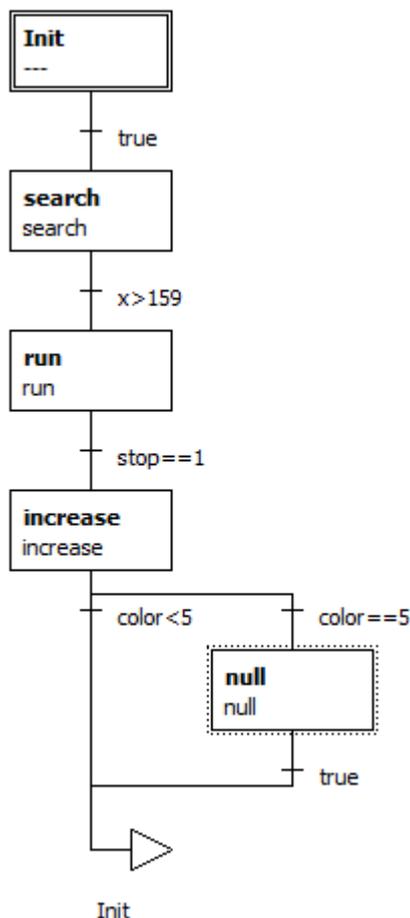


Рисунок 1 – Блок-схема главной программы

На шаге **search** (рис. 2) робот производит поиск текущего цвета сегмента, совершая вращательное движение с заданной угловой скоростью.

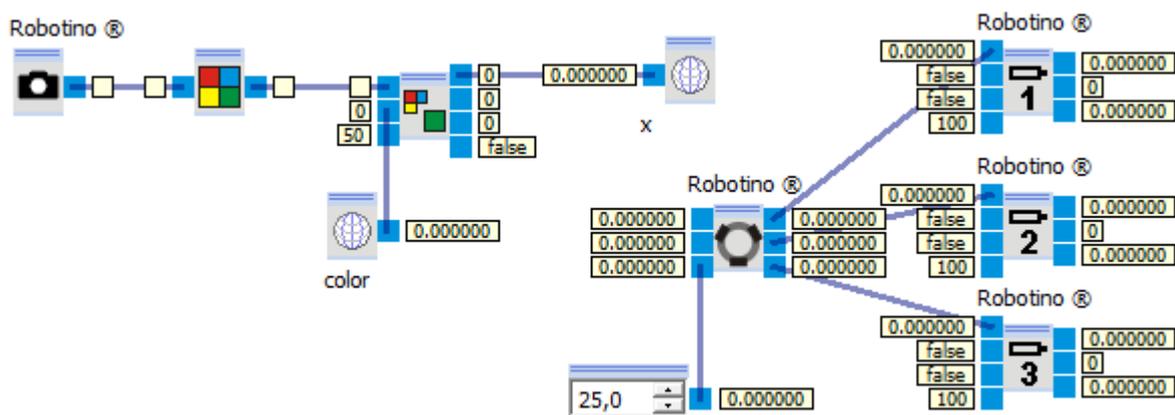


Рисунок 2 – Структура шага поиска цвета сегмента (search)

В блоке Segmenter (рис. 3) при включенной симуляции присваиваются порядковые номера цветов всех сегментов. Для этого, во время вращения робота, при попадании в область видимости камеры нужного цвета ставиться флажок Freeze image, затем цвет выделяется мышкой и добавляется в директиву кнопкой '+'. Аналогично для других цветов.

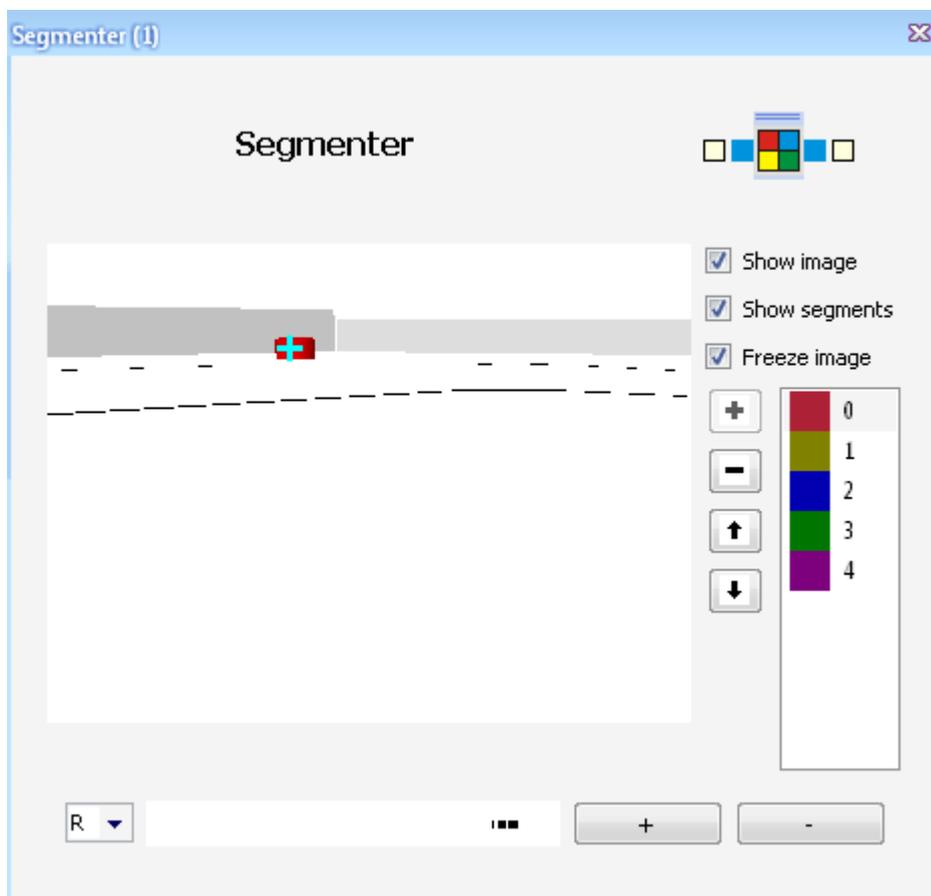


Рисунок 3 – Содержимое блока Segmenter

Изображение найденного сегмента с текущим порядковым номером цвета подводится к Segment Extractor (рис. 4) к сигналу Input (Image).

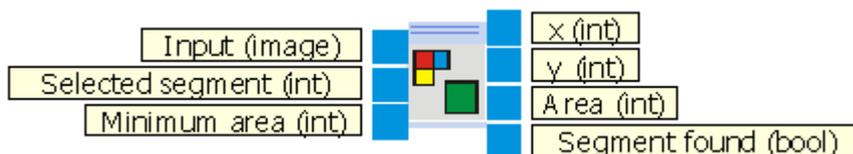


Рисунок 4 – Описание входных и выходных сигналов блока Segment Extractor

Input (image) – входящее изображение с Segmenter; Selected segment (int) – порядковый номер текущего цвета сегмента; Minimum area (int) – минимальная площадь сегмента для обнаружения; x (int) – текущая координата положения центра сегмента по оси X; y (int) – текущая координата положения центра

сегмента по оси Y; Area (int) – текущая площадь сегмента; Segment found (bool) – логическая переменная, которая сообщает, что сегмент найден

Канал Selected segment считывает текущий порядковый номер сегмента с переменной **color**. Текущая координата положения центра сегмента по оси X записывается в переменную **x**.

На следующий шаг **run** программа переходит при условии, если  $x > 159$ , где 159 – это половина ширины входящего изображения с камеры (в нашем случае 320 пикселей). Для камер с другим расширением эта величина будет определяться по формуле (1):

$$\frac{\text{ширина изображения (в пикселях)}}{2} - 1 \quad (1)$$

В шаге **run** (рис. 5) константой задается скорость движения по оси X и сопоставляется центр изображения камеры с центром текущего сегмента, регулируя тем самым положение робота относительно оси Y.

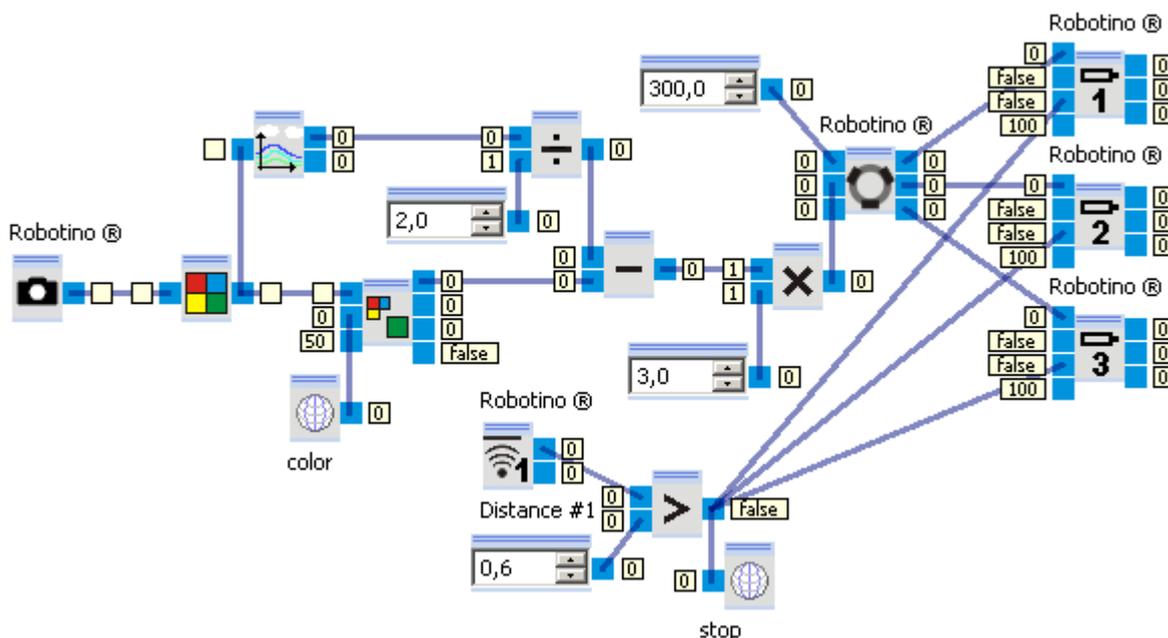


Рисунок 5 – Структура шага движения робота к текущему цвету сегмента

По мере приближения робота к объекту, на датчик поступает сигнал. Когда сигнал с датчика превышает заданную константу, результат сравнения записывается в переменную **stop**. Условием перехода на следующий шаг является равенство этой переменной значению 1 (true).

В шаге **increase** (рис. 6) к текущему значению **color** прибавляется единица, тем самым меняя искомый цвет.

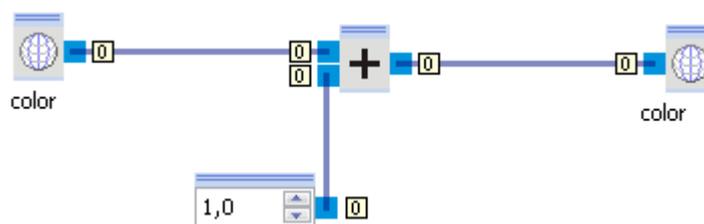


Рисунок 6 – Структура шага изменения искомого цвета сегмента

Затем в main program происходит разветвление по двум условиям. Если порядковый номер цвета меньше 5 (число заданных цветов), то программа возвращается в начало и выполняет вышеуказанные шаги для следующего цвета. Если порядковый номер цвета равен 5, то программа попадает на шаг **null** (рис. 7), который обнуляет переменную **color**. Затем программа производит повторный поиск начального цвета.

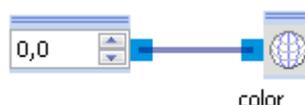


Рисунок 7 – Структура шага обнуления цвета сегмента

### Задание на самостоятельную работу

1. Составить программу по примеру, проверить работоспособность в Robotino Sim или на реальном роботе.
2. Составить свою программу, поменяв порядок поиска цветов роботом.