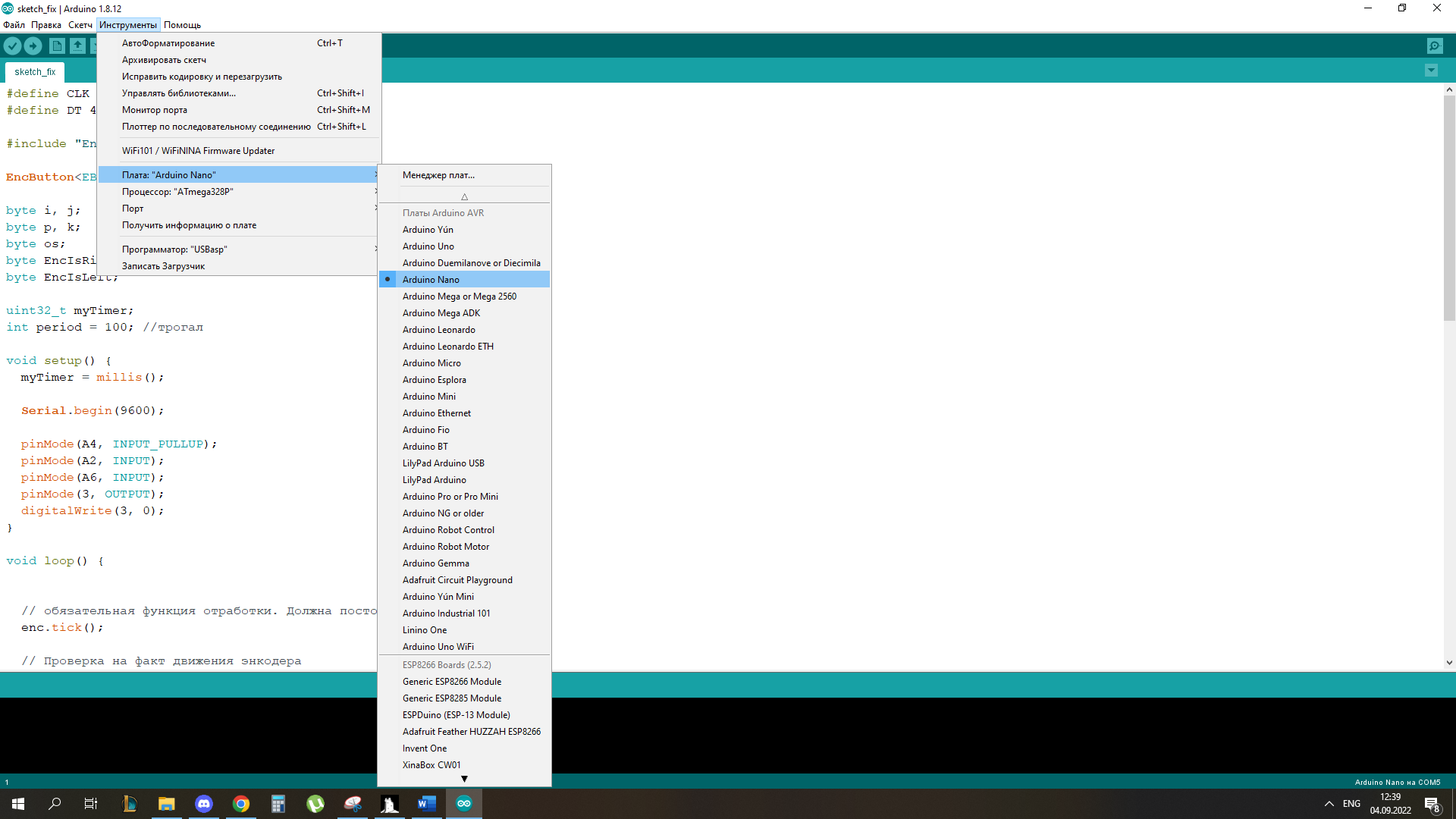
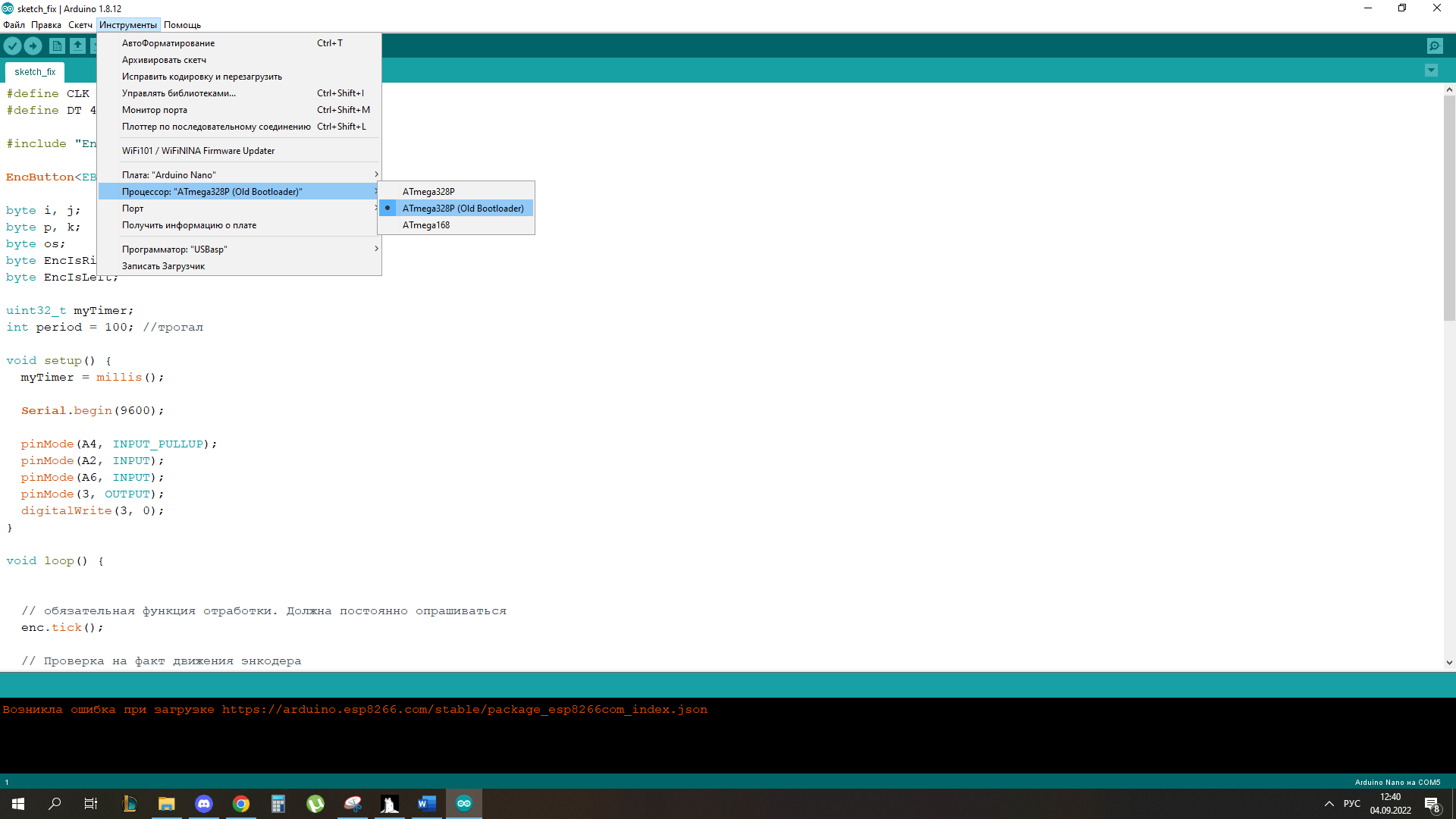
**Руководство по использованию**

Данную тестовую версию разработал Юхно Виктор Витальевич, почта для связи: [yvv0498@gmail.com](mailto:yvv0498@gmail.com)

Для начала использования в ПО Arduino IDE, во вкладке «Инструменты» - выбираем плату Arduino Nano, выбираем процессор ATMEGA 328P old boot.



Затем, для использования библиотеки обработки энкодера – заходим во вкладку «инструменты», «управлять библиотеками», в поиске ищем «EncButton», устанавливаем.

**Краткое описание программы**

Программа предоставлена с комментариями, упрощающими чтение программы сторонними разработчиками. Протестирована для различных задач как источник сигналов моделирования. Спорным функционалом остается включение индикатора – светодиода, а также – ограниченность функционала энкодера. В данный момент передается только факт поворота «налево», «направо», без встроенного счетчика. Библиотека «EncButton» упростит реализацию возможного функционала с использованием счетчика положения.

Пример распиновки в текущей физической сборке:

* D4 DT – энкодер
* D3 – светодиод
* D2 CLK – энкодер
* A2 – потенциометр
* A6 – потенциометр
* A4 - кнопка

Объявляем переменные, и связанные с ардуино порты.

CLK 2 DT 4 – порты связанные с энкодером, один отвечает за поворот по часовой стрелке, второй против часовой. Так как физическое подключение кнопки энкодера – отсутствует, в данной версии ПО – она не используется.

Байтовые переменные необходимы для обмена данными с по SimInTech

Переменные i,j – связанны с кнопкой и светодиодом. Переменные p,k – связанны с потенциометрами. Переменная os – связанна с получением данных из порта.

Переменные EncIsRight, EncIsLeft – нужны для передачи данных о повороте энкодера в одну из сторон.

Serial.begin(9600) – отвечает за скорость обмена данных с портом.

Фрагмент программы:

p = map(analogRead(A2), 0, 1023, 0, 255);

k = map(analogRead(A6), 0, 1023, 0, 100);

Необходим для масштабирования данных. Физическое, необработанное представление данных происходит в формате 0 – 1023, где 0 крайне левое положение потенциометра, 1023 соответственно крайне правое. (Никаких политических координат).

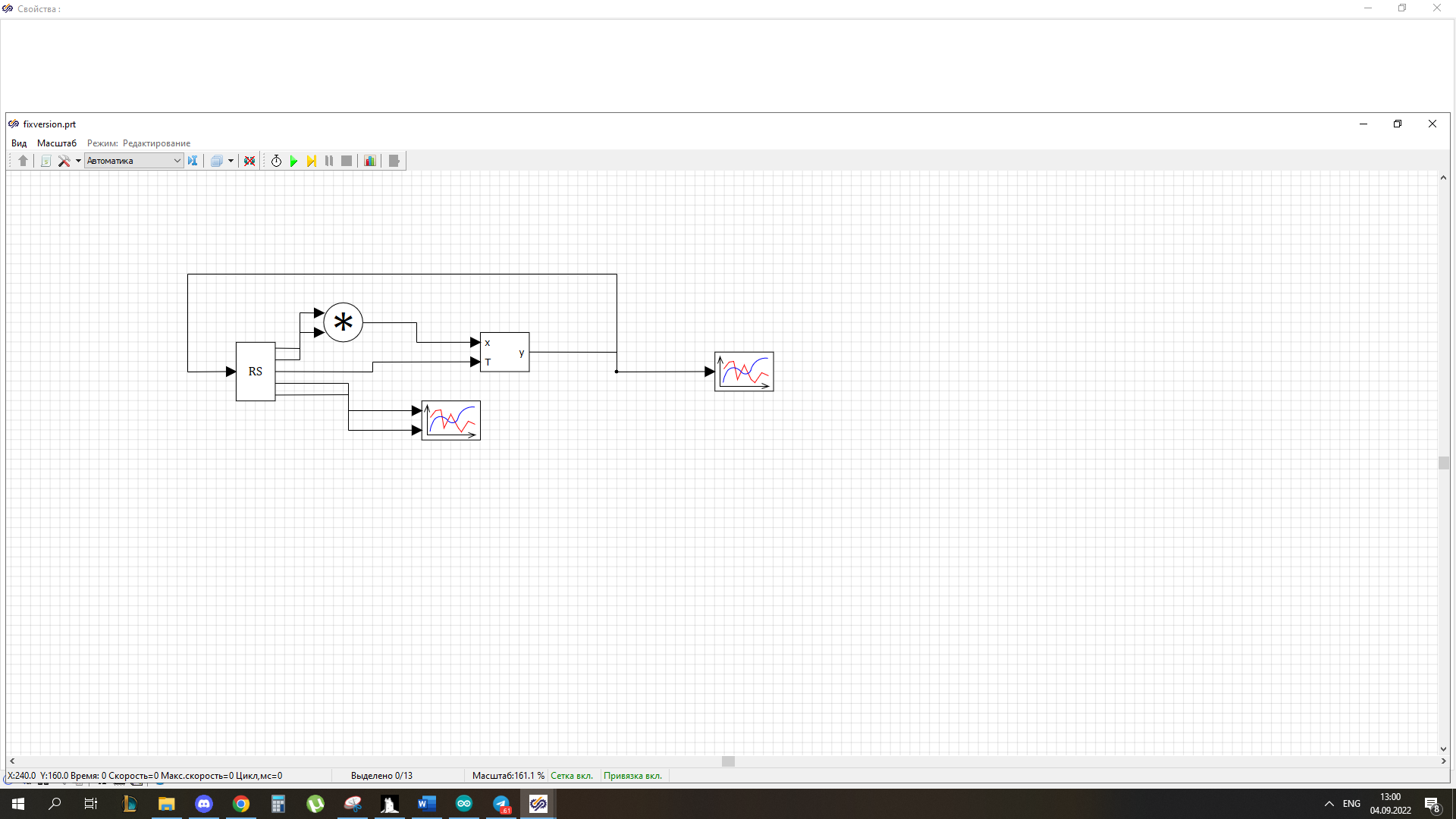
Функция map пропорционально преобразует данные значения, таким образом, крайне левое положение представляет значение 0 (можно задать любое, отличное от 0), крайне правое, в случае A2 – 255. Если будет необходимость в ходе лабораторных работ использовать другие значения с потенциометров – задается именно в этом фрагменте кода.

int period = 100;

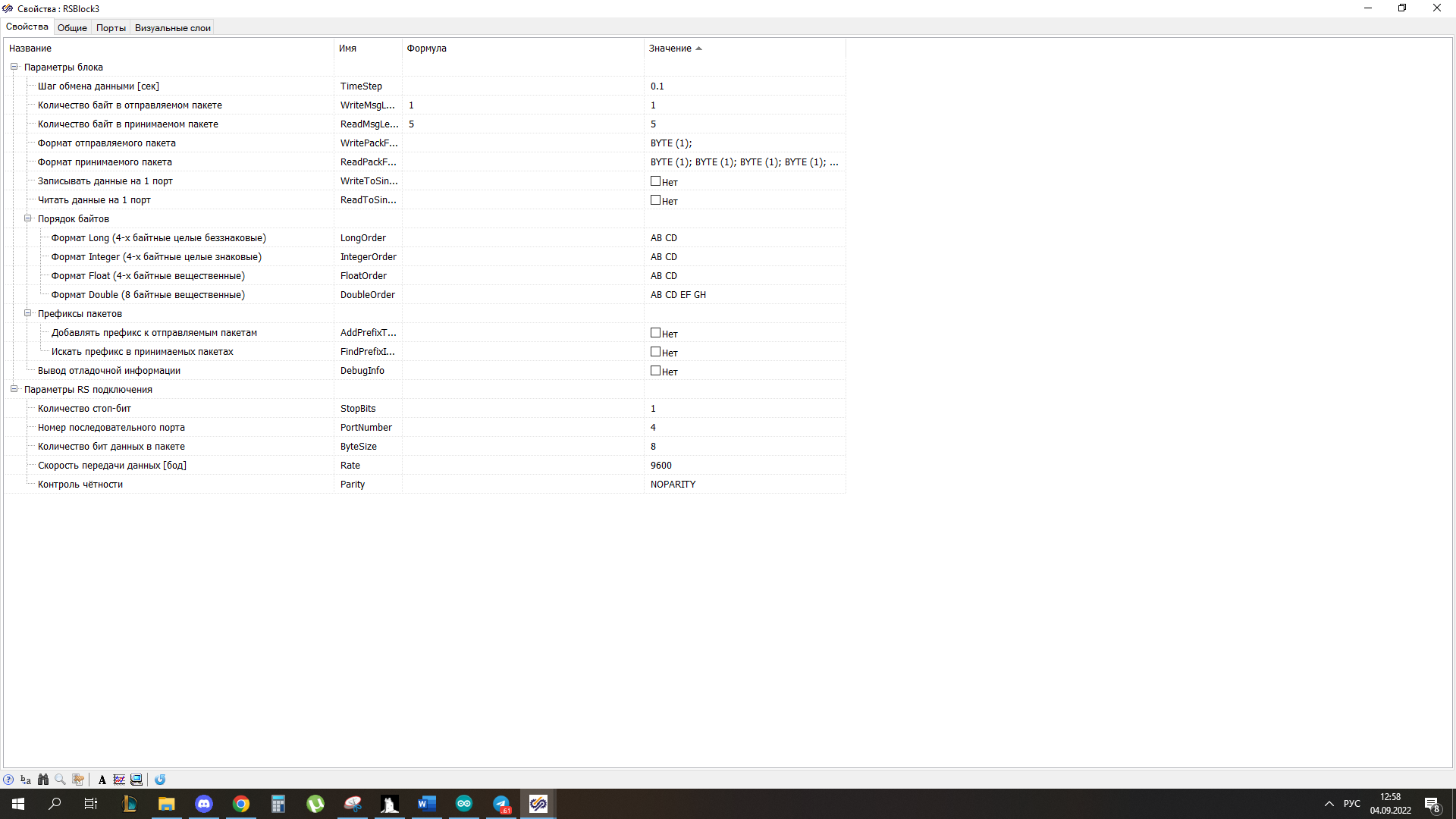
Отвечает за внутренний параметр таймера, при некорректной связи – можно попробовать изменить значение в меньшую или большую сторону.

**Настройка в SimInTech**

Для обмена данными с портом необходимо использовать блок RS приведенный на скриншоте ниже.



Параметры блока RS приведены на скриншоте ниже:



Шаг обмена данными должен соответствовать параметру int period = 100;

Однако, при работе с различными портами это значение **может** отличаться, в таком случае – его значение следует подобрать эмпирическим путем. (Иногда при сложной программе в симентеке происходит рассинхрон, и ардуина не вывозит переваривать обмен данных и вовремя попадать в такты. В связи с этим может происходить некорректная передача пакета данных. Фиксить либо переписыванием кода с нуля, с учетом будущих требований, либо смириться с пониженной частотой обмена данными.)

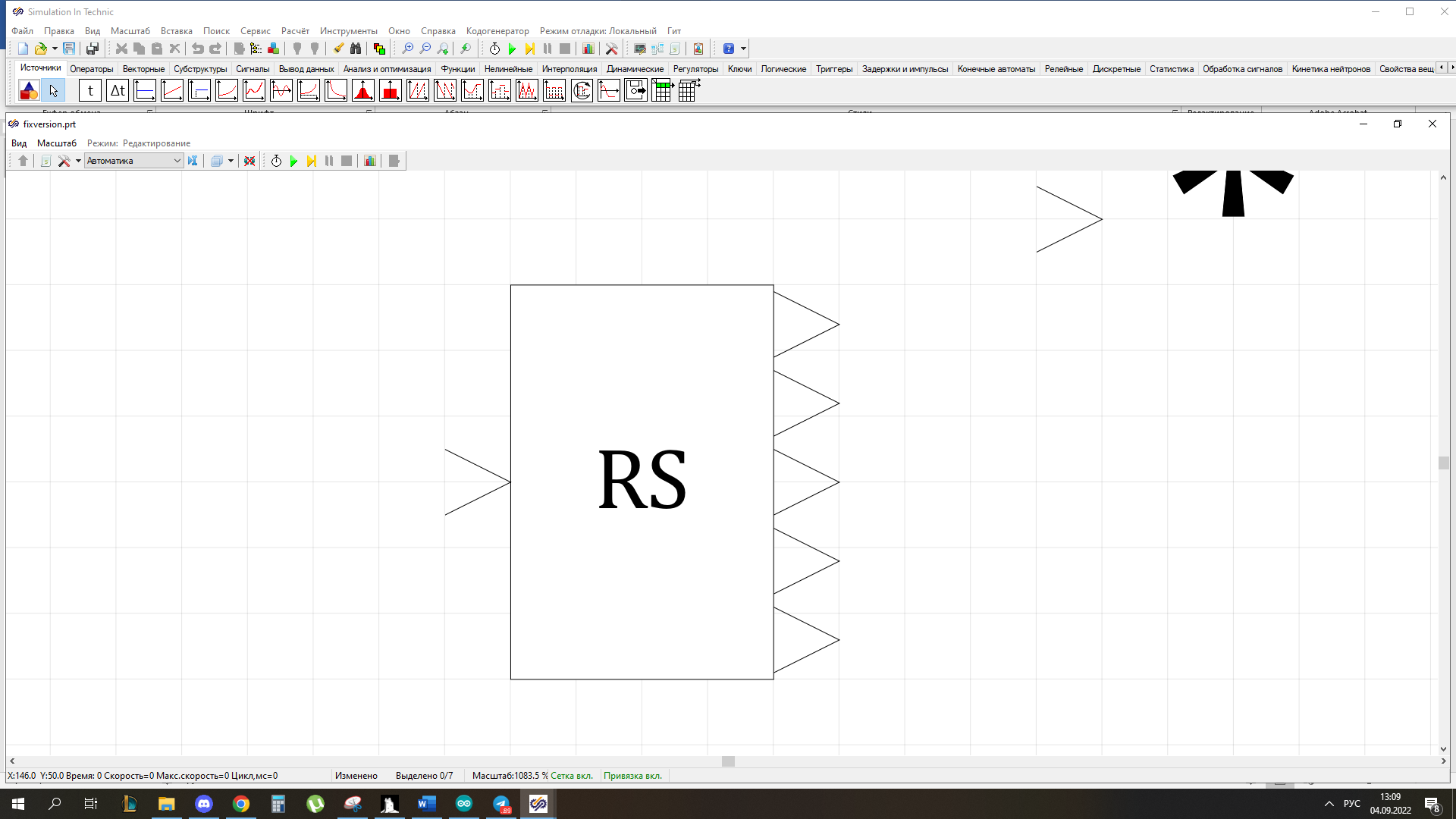
Количество байт в отправляемом пакете – 1, так как мы передаем только факт включения/выключения светодиода. (Этот фрагмент кода прошивки тоже следует переписать при использовании, во время других задач)

Принимаемый пакет – устанавливаем соответствие согласно порядку и количеству передаваемых данных. В текущем примере все данные однобайтовые.

Номер порта – номер последовательного порта для связи с контроллером.

Скорость передачи данных **ВСЕГДА** должна соответствовать скорости в прошивке.

После всех настроек получаем блок данного вида.



Соответствие портов в правой части RS блока:

1. Serial.write(i);
2. Serial.write(p);
3. Serial.write(k);
4. Serial.write(EncIsRight);
5. Serial.write(EncIsLeft);

Слева блок соответствующий функции os = Serial.read();