ГЛОНАССSoft

www.glonasssoft.ru

MyL<>gic

Технология быстрого создания и внедрения индивидуальных алгоритмов работы терминалов УМКа

01 Построение нестандартной логики устройства

02 Поддержка специфического и редко используемого оборудования

Индивидуальный подход к решению задач вашего бизнеса!

+60 реализованных кейсов

- гибкая система, за счет использования скриптового языка
- помощь в написании скрипта
- бережное использование памяти и прочих ресурсов
- написание скрипта под заказ

Будем рады сотрудничеству с вами!
По всем вопросам обращайтесь:

Тел: 8 800 700 82 21
e-mail: info@glonasssoft.ru

www.glonasssoft.ru

ГЛОНАССSoft

www.glonasssoft.ru

MyLogic

Какие терминалы УМКа поддерживают технологию MyLogic:

- **УМКа303** - максимальный набор функций включая работу с камерами, CAN-шиной, геозонами, интерфейсами и BLE
- **УМКа302** - аналогичный набор функций за исключением работы с камерами
- **УМКа310/311/312** - базовый набор функций включающий работу с навигацией, интерфейсами и BLE
- **УМКа300/301** - старые модели терминалов не поддерживают работы с MyLogic

Будем рады сотрудничеству с вами!
По всем вопросам обращайтесь:

Тел: 8 800 700 82 21
e-mail: info@glonasssoft.ru

www.glonasssoft.ru

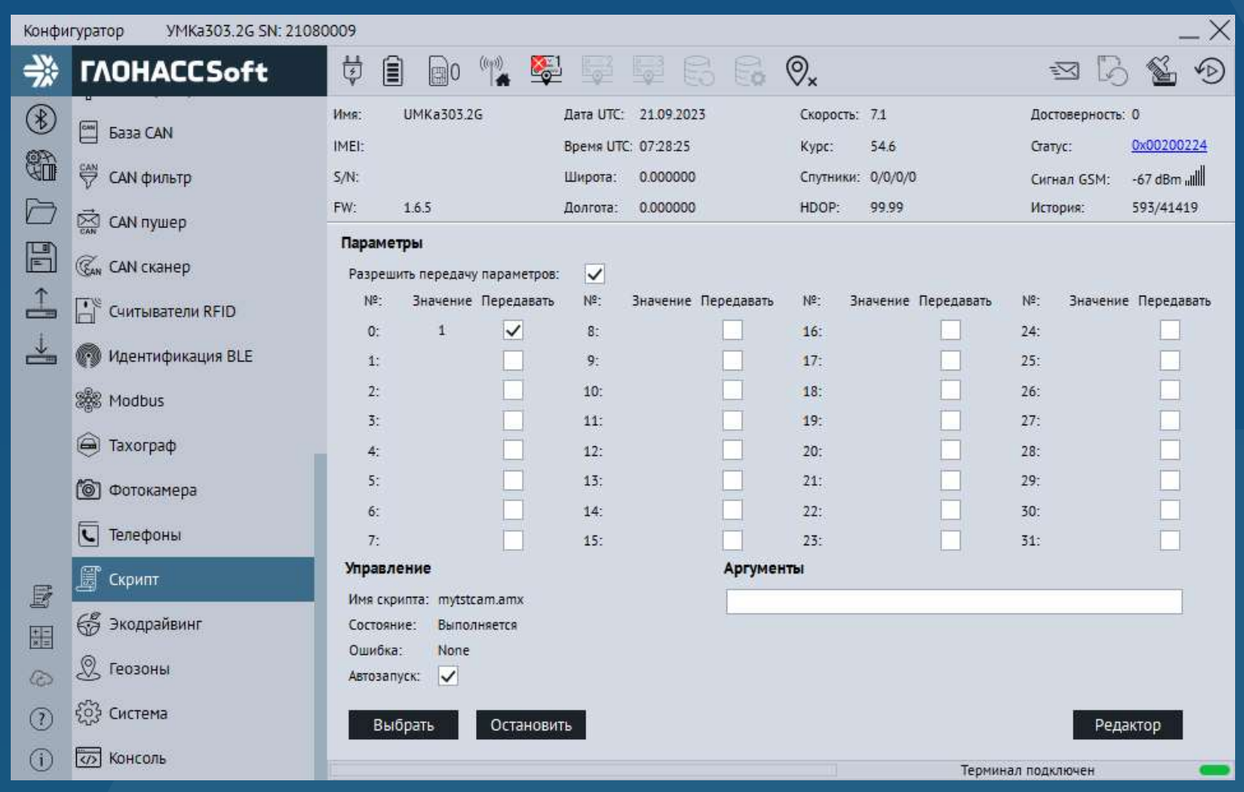
ГЛОНАССSoft

www.glonasssoft.ru

MyLogic

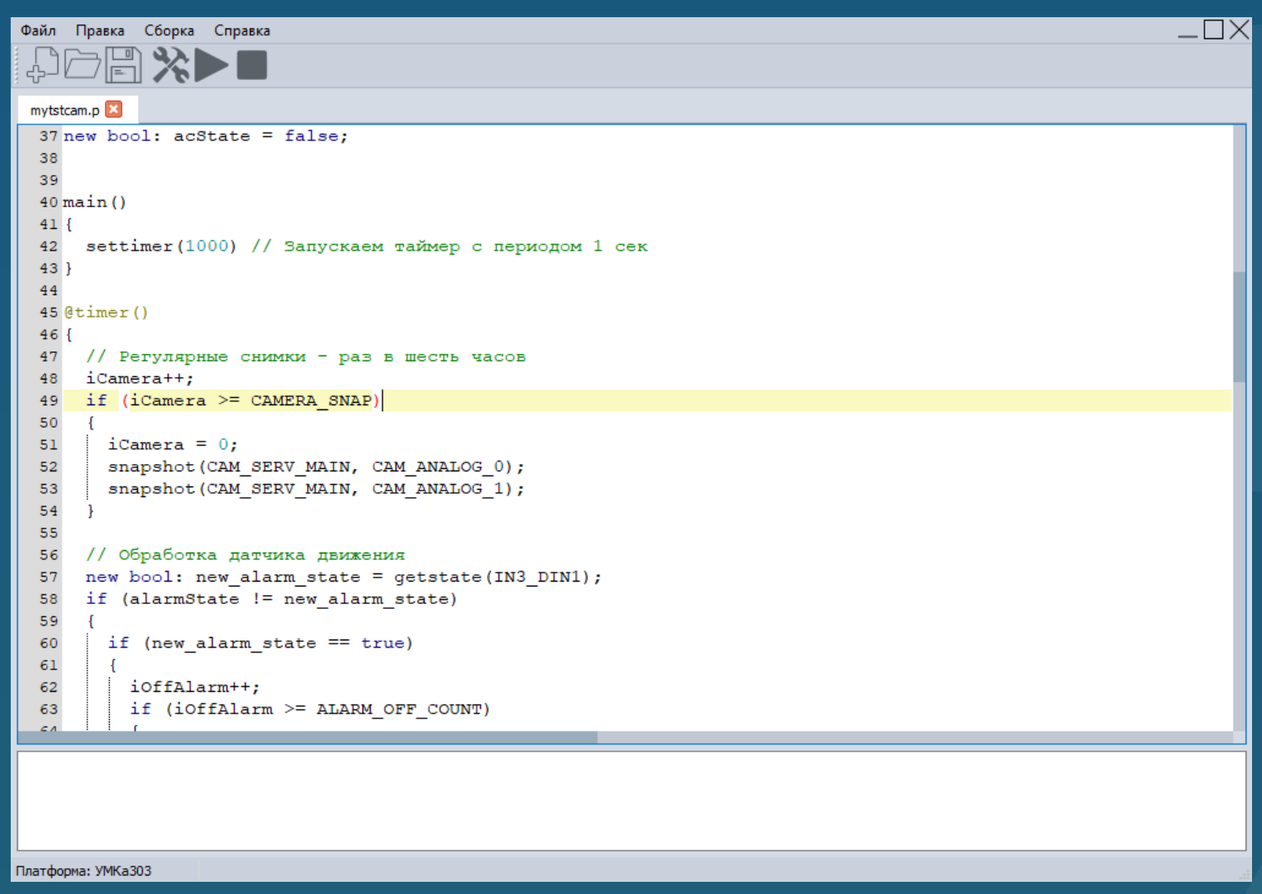
Работа со скриптами в Конфигураторе УМКа3хх

Управление скриптами MyLogic



- **Легкий интерфейс**
- **Гибкие возможности**
- **Удобное управление**

Редактор скриптов MyLogic



- **Весь необходимый функционал**
- **Удобный формат**
- **Быстрый запуск скрипта**

Будем рады сотрудничеству с вами!
По всем вопросам обращайтесь:

Тел: 8 800 700 82 21
e-mail: info@glonasssoft.ru

www.glonasssoft.ru

ГЛОНАССSoft

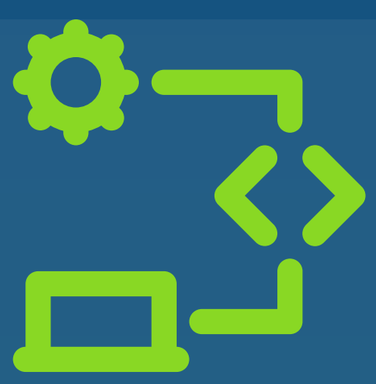
www.glonasssoft.ru

MyLogic

Архитектура технологии

Для написания скриптов MyLogic используется простой, не типизированный 32-битный скриптовый язык программирования Pawn с Си-подобным синтаксисом.

Основные части набора инструментов PAWN:

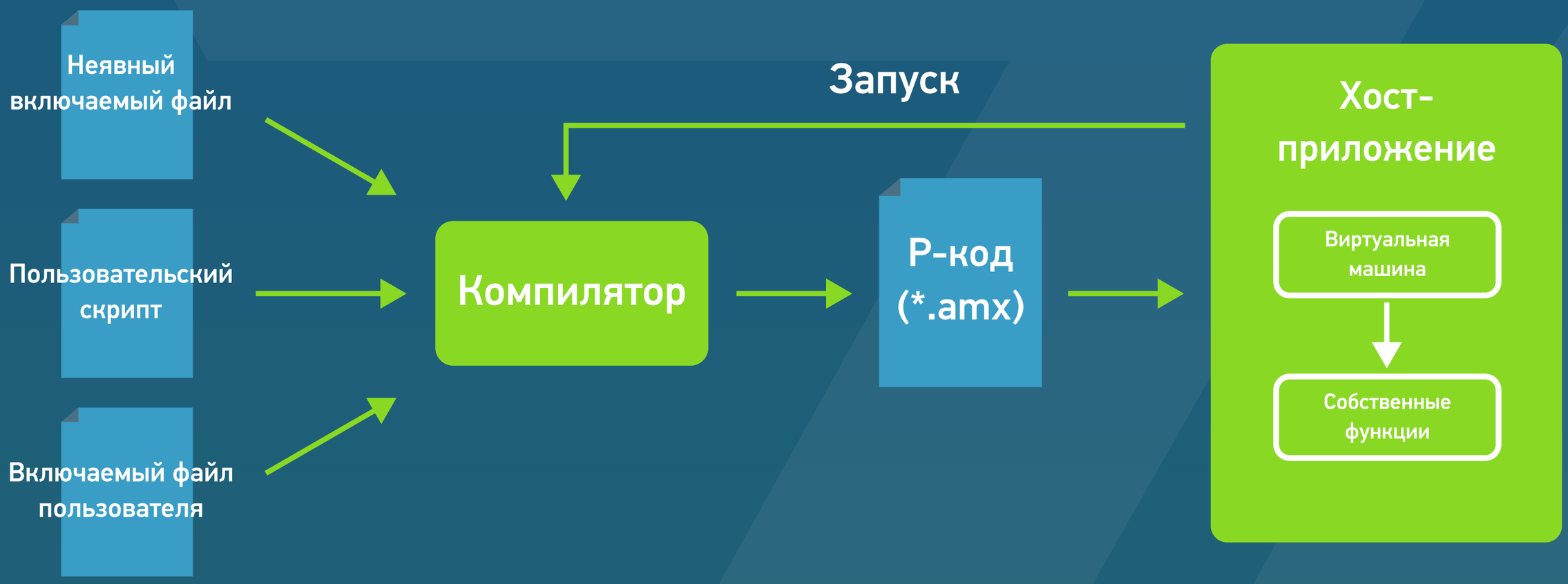


Компилятор

Виртуальная машина

Как это работает?

Скрипт компилируется в байт-код, который запускается внутри виртуальной машины.



Будем рады сотрудничеству с вами!
По всем вопросам обращайтесь:

Тел: 8 800 700 82 21
e-mail: info@glonasssoft.ru

www.glonasssoft.ru

MyLogic

Создание скрипта

Си подобный подход

Заключается в использовании “входной” функции “main”

Возможно использование Arduino-подобного подхода с функциями @setup и @loop

```
1
2  main()
3  {
4    new CountEntered[COUNT_PP1];
5    new CountOut[COUNT_PP1];
6    new StatDoor[COUNT_PP1];
7    new OldStatDoor[COUNT_PP1];
8
9    new bool:Valid
10   new bool:isEvent
11   new Addr
12   new i
13   isEvent = false
14
15
```

Объявление переменных

В языке MyLogic существует несколько типов переменных:

- Целочисленные
- Числа с плавающей точкой
- Логические

По области видимости:

- Глобальные
- Локальные

Именованные ячейки памяти для хранения данных

```
1
2  //Для объявления новой переменной
3  используется оператор “new”
4
5
6   new Var = ...;
7
8   new Float: Var = ...
9
10  new bool: Var = true/false
11
12  const IntConst = 100;
```

Будем рады сотрудничеству с вами!
По всем вопросам обращайтесь:

Тел: 8 800 700 82 21
e-mail: info@glonasssoft.ru

MyLogic

Создание скрипта

Массивы

Именованная область памяти, предназначенная для хранения нескольких переменных одного типа

По количеству используемых индексов массивы различаются:

- Одномерные
- Двумерные

```
1
2  new IntArray2[3] = [1210,1220,1230] // Одномерные
3
4  new Int2Array = [10] [20] // Двумерные
5
6
7  new CharArray {3} = {121, 122,123}; // Объявление
8
9  new CharArray {1} = 128; // Пример обращения
10
```

Для объявления массива побайтовым доступом используются фигурные скобки

Операторы

Элемент языка, задающий описание действия, которое необходимо выполнить

Все операторы MyLogic, условно разделены на следующие категории:

- Математические операторы
- Операторы сравнения
- Логические операторы
- Условные операторы
- Операторы цикла
- Операторы перехода

```
1  // Пример использования операторов
2  new a = 3;
3  new b = 25;
4  if (a<b) // Если “a” меньше “b”
5  {
6      printf (“a less b”) ; // Выводим сообщение “a меньше b”
7  }
8  else if (a == b) // Иначе если “a” равно “b”
9  {
10     printf (“a eguals b”) ; // Выводим сообщение “a  равно b”
11 }
12 else //Иначе
13 {
14  printf (“a more b”) ; // Выводим сообщение “a  больше b”
15 }
```

ГЛОНАССSoft

www.glonasssoft.ru

MyL<>gic

Создание скрипта

Функции

Функции в скриптовом языке условно можно разделить на следующие виды:

- Функции, непосредственно реализованные в скрипте;
- Библиотечные функции;
- Функции платформы (нативные функции);
- Функции скрипта, вызываемые платформой (callback функции);

Именованная последовательность операторов, которая определена и записаны только в одном месте скрипта.

```
1
2   // Синтаксис
3   <имя> (<параметр>)
4   {
5   <действие1>;
6   <действие2>;
7   <действиеN>;
8   <return value>
9   }
10
```

Событийная модель

Парадигма при которой выполнение скрипта определяется системными событиями.

Основные события:

@timer - срабатывание таймера
@accupdate - новые данные от акселерометра
@gnssupdate - новые данные от GNSS приемнника
@bleadvertrecv0-3 - поступление данных от BLE
@canrecv - поступление данных по CAN-шиине
@chat - поступление специальной команды
@bboxupdate - добавление точки в черный ящик

Пример обработки события каждую секунду

```
1   # include <tracker>
2   # include <time>
3
4   # pragma dynamic 32
5
6   new CountEventTimer = 0;
7
8   main ()
9   {
10      settimer (1000) // Запускаем таймер с периодом 1 сек.
11  }
12
13  @timer ()
14  {
15      CountEventTimer++; // Увеличиваем значение счетчика
    событий таймера
16      printf (“Timer event #%d”,  CountEventTimer) // Выводим в
    отладочный вывод
17  }
18
19
```

Будем рады сотрудничеству с вами!
По всем вопросам обращайтесь:

Тел: 8 800 700 82 21
e-mail: info@glonasssoft.ru

MyL<>gic

Создание скрипта

Передача параметров на сервер

Все параметры, предназначенные для передачи на сервер в устройствах УМКа3ХХ сохранятся в чёрном ящике устройства.

- платформа позволяет передавать до 32х параметров (тэгов)
- значение одного тега - 32 бита, это может быть целое, дробное или логическое значение.
- для каждого параметра может быть задан признак валидности

settag - функция производит запись значения в ячейку для передачи на сервер

pushpoint - функция принудительно формирует точку в черном ящике

Отладка скрипта

Для вывода отладочных сообщений из скрипта используется функция printf(). Вывод сообщений производится непосредственно в консоль конфигуратора.

Директивы препроцессора

#if - #endif данная директива говорит компилятора какие фрагменты исходного кода необходимо компилировать а какие пропустить

#define позволяет вводить в текст программы константы и макроопределения
#define <идентификатор> <замена>

#include подключает в скрипт ранее созданные библиотеки

ГЛОНАССSoft

www.glonasssoft.ru

MyLogic

Создание скрипта

Библиотеки платформы MyLogic

- ble.inc - библиотека содержит основные функции работы с BLE-устройствами и метками включая расстояние
- modem.inc - библиотека содержит функции работы с SMS-сообщениями модема и входящими звонками
- args.inc - библиотека содержит функции работы с параметрами запуска скрипта
- time.inc - библиотека содержит функции работы с временем и таймерами

Совместимость платформ и библиотек

Библиотека	УМКа302	УМКа303	УМКа310	УМКа311	УМКа312
tracker.inc	+	+	+	+	+
geofence.inc	+	+			
serial.inc	+	+	+	+	+
can.inc	+	+			
ble.inc	+	+	+	+	+
modem.inc	+	+	±	±	±
args.inc	+	+			
time.inc	+	+	+	+	+

Будем рады сотрудничеству с вами!
По всем вопросам обращайтесь:

Тел: 8 800 700 82 21
e-mail: info@glonasssoft.ru

www.glonasssoft.ru