

Математика и информатика в образовании и бизнесе. Сборник материалов международной научно-практической конференции. (г.Москва, апрель 2020 г.). - М.: Aegitas, 2020. – С.39-44.

Бардынин Дмитрий Николаевич,

магистр 1-го курса направление «Педагогическое образование», профиль «Развитие детской одаренности», Дирекции образовательных программ, ГАОУ ВО МГПУ

Научный руководитель: Йоффе Андрей Наумович,

профессор, доктор педагогических наук, профессор кафедры педагогических технологий непрерывного образования, Института непрерывного образования ГАОУ ВО МГПУ

ПРАКТИКА ПРИМЕНЕНИЯ АЛГОРИТМИЧЕСКОГО ЯЗЫКА «ДРАКОН» ДЛЯ ПРОГРАММИРОВАНИЯ УСТРОЙСТВ НА БАЗЕ ПЛАТФОРМЫ ARDUINO

Bardynin Dmitriy Nikolaevich

First year magister of Pedagogical education, profile Development of children's talent, Directorate of educational programs, Moscow City University

Scientific supervisor: Ioffe Andrey Naumovich,

Professor, Professor, Doctor of Pedagogical Sciences, Professor of the Department of Pedagogical Technologies of Continuing Education, Moscow City University

PRACTICE OF USING THE "DRAKON" ALGORITHMIC LANGUAGE FOR PROGRAMMING DEVICES BASED ON THE ARDUINO PLATFORM

Аннотация: Статья посвящена писанию практики применения визуального алгоритмического языка программирования и моделирования ДРАКОН в образовательном центре «Сириус» г.Сочи. Автором приведен способ расширения круга пользователей информационных технологий в рамках образовательной проектной деятельности.

Abstract: The article is devoted to the description of the practice of using the visual algorithmic programming and modeling language DRAKON in the educational center “Sirius” in Sochi. The author provides a method for expanding the range of information technology users in the educational project activities.

Ключевые слова: ДРАКОН, визуальное алгоритмическое программирование, ARDUINO

Keywords: DRAKON, visual algorithmic programming, ARDUINO

Существующая практика ведения занятий со школьниками с использованием классического текстового программирования, с использованием распространенных языков типа C++, Python, Java, или менее распространенных Pascal, Basic, Кумир, сталкивается с существенной проблемой. Для того, чтобы грамотно решить предложенную задачу, необходимо предварительно создать модель поведения исполнителя, который эту задачу будет решать. Если в простом упражнении, содержащем несколько математических или строковых операций, подобная модель поведения довольно простая, линейная, логическая древовидная или цикличная, и зачастую эта модель даже не осознается автором, то в сложном проекте, где решается множество задач, часто синхронно и взаимозависимо, построение модели поведения жизненно необходимо.

ФГОС определяет серьезные требования к образовательному процессу, стимулируя проведение учащимися проектной деятельности, что принесет безусловную пользу только в случае создания уникальных проектов, способных раскрыть творческий потенциал каждого школьника. Если мы говорим об информационных технологиях, то критически важно предоставить учащимся инструменты не только для самовыражения, но и для качественного обмена информацией с педагогами, товарищами и неограниченным кругом единомышленников. Выбор одного из существующих языков программирования существенно ограничивает круг общения, определяя его людьми, способными понять содержание проекта, т.е. владеющими языком, примененным в проекте. Помимо ограниченности круга общения, формальные правила языка программирования могут ограничить сам процесс творчества, требуя соблюдения требований синтаксиса, построения допустимых управляющих конструкций, написания корректных формул, что потребует значительных усилий от проектанта, порою не готового к такого рода деятельности в силу малого интереса именно к языкам программирования, а не к предмету проектирования.

Дополнительным ограничением для создания качественного проекта является ограниченный объем учебных часов, которые можно потратить на работу над проектом. Эта ситуация особенно заметна в образовательных центрах типа «Сириус», где дети круглосуточно находятся под опекой педагогов и кураторов, и их график насыщен разнообразными событиями.

В центре дополнительного образования «Сириуса» направление «Алгоритмизация и программирование» дает возможность детям с различным уровнем подготовки реализовать собственные проекты в сжатые сроки на базе популярной у любителей микроконтроллерной техники платформы Arduino.

Преподавание осуществляется как среди одаренных участников образовательных программ «Сириуса», так и среди обычных школьников, попадающих в центр дополнительного образования в рамках программ работы со школами г.Сочи. Также участниками программы становятся дети из других городов, областей, стран, посещающие Центр в период школьных каникул, и не только.

Программирование микроконтроллерных устройств производится с использованием комплекса ПО, состоящего из редактора языка ДРАКОН Степана Митькина, доработанного для использования в составе связки с Arduino IDE и Processing IDE, генераторов скриптовых файлов, вспомогательных программ. Создание комплекса позволило в короткий срок

подготовить учащихся к процессу проектирования, не отвлекаясь на работу с текстами создаваемых программ.

Язык ДРАКОН [2] представляет собой графический язык, позволяющий с помощью ограниченного числа графических элементов описать модель поведения достаточно сложных систем, не прибегая к глубокому погружению в программирование, как таковое.

В состав языка входят классические элементы блок-схем алгоритмов, описанных в ГОСТ 19.701-90 и ISO 5807-85. Важным элементом языка являются правила построения диаграмм [1], корректное использование которых может облегчить понимание диаграмм даже неподготовленным читателем. При создании языка применялись принципы когнитивной эргономики, позволяющие воспринимать визуальные алгоритмы симультанно, а не сукцессивно, что неизбежно при использовании текстовых способов, или визуальных языков программирования типа Scratch.

Редактор языка ДРАКОН позволяет создать понятную схему поведения устройства на микроконтроллере, причем для этого достаточно довольно поверхностных знаний о языке программирования, применяемого в проекте. В качестве иллюстрации можно рассмотреть лабораторную работу по измерению ускорения свободного падения на машине Атвуда [3].

Для примера приведен текст программы на языке Wire, которая позволяет использовать Arduino для обеспечения проведения эксперимента (см.рис.1).

```
const int buttonPin = 9;
const int MagnetPin = 6;
int buttonState = 0;
int incomingByte;
void setup() {
  Serial.begin(9600);
  pinMode(MagnetPin, OUTPUT);
  pinMode(buttonPin, INPUT);
}
void loop() {
  incomingByte = Serial.read();
  if (incomingByte == 'O') {
    digitalWrite(MagnetPin, HIGH);
  }
  if (incomingByte == 'F') {
    digitalWrite(MagnetPin, LOW);
  }
  buttonState = digitalRead(buttonPin);
  if (buttonState == HIGH) {
    Serial.print(1, DEC);
    delay(3000);
  }
}
```

Рис. 1 – Пример текста программы на языке Wire лабораторной работы по измерению ускорения свободного падения на машине Атвуда.

Приведенный на рис.1 текст будет понятен читателю лишь после предварительной подготовки, если он уже знаком с языком Wire, или с каким-то языком типа Си.

Если заменить текстовые команды языка на графические элементы, расположенные в соответствии с правилами языка ДРАКОН, то читабельность алгоритма существенно повышается, и его можно представлять лицам, с программированием никак не связанными, но хорошо представляющими себе предмет проектирования, в предлагаемом примере это и машина Атвуда, и сам физический эксперимент.

Приведенный на рис.2 алгоритм не является исчерпывающим описанием, позволяющим сгенерировать программный код для платы Arduino. Применяемая технология генерации предусматривает выбор языка генерируемого кода, и в принципе мы можем создать программу на многих языках, входящих в набор редактора ДРАКОНа, например, на языке Кумир, С, С#, Python, Java, Javascript, TCL, Erlang, Lua, Verilog и др. Единственным обязательным условием является корректное формирование базовых элементов построения целевого алгоритма, которые необходимо определить по правилам синтаксиса выбранного языка генератора.

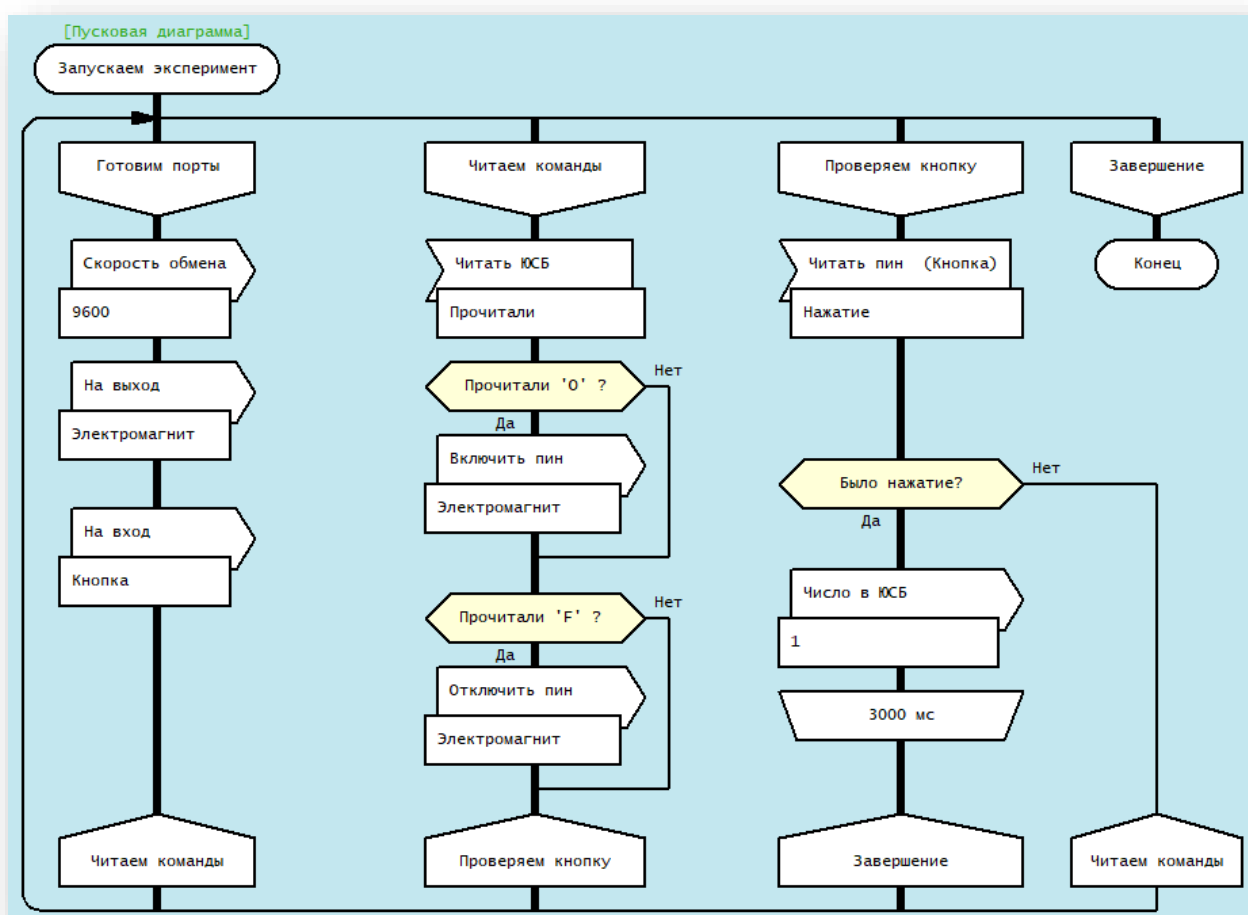


Рис. 2 – Пример оформления алгоритма лабораторной работы по измерению ускорения свободного падения на машине Атвуда.

Применяемый подход является примером создания программных продуктов без использования специальных знаний, присущих программистам, и в полной мере может служить примером реализации подхода «Программирование без программистов».

Данный подход был апробирован на различных группах учащихся, разных возрастов, с разной степенью подготовки и мотивации. Дети, ранее не использовавшие информатику для решения повседневных задач, с успехом создавали проекты, привязанные к их сфере интересов: спортсмены проектировали спортивные тренажеры, скульпторы разрабатывали подвижные модели, математики проектировали роботов со сложной кинематикой.

Использование визуальных алгоритмов на языке ДРАКОН позволило упростить вхождение в информационные технологии людям с минимальными знаниями о программировании и микроконтроллерной технике. Примеры из привычной среды, бытовые правила и шаблоны поведения, оформленные по правилам языка ДРАКОН, снизили порог вхождения в целевую область знаний, и позволили связать знакомые образы с новыми технологиями.

Список литературы

1. Паронджанов В. Д. Дружелюбные алгоритмы, понятные каждому. Как улучшить работу ума без лишних хлопот. — М.: ДМК-пресс, 2010. — 464 с.
2. Паронджанов В. Д. Занимательная информатика \ Учебное пособие для учащихся 5–9 классов. — М.: Дрофа, 2007. — 192 с.
3. Минкин А.В., Дерягин А.В., Ибатуллин Р.Р. ИСПОЛЬЗОВАНИЕ МИКРОКОНТРОЛЛЕРА ATMEGA32 НА УРОКАХ ФИЗИКИ // Научное обозрение. Педагогические науки. – 2015. – № 2. – С. 159-160