

**РАЗРАБОТКА АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ С ИСПОЛЬЗОВАНИЕМ
ОБЛАЧНЫХ ТЕХНОЛОГИЙ И АРДУИНО****DEVELOPMENT OF AN AUTOMATED SYSTEM USING THE CLOUD TECHNOLOGY
AND ARDUINO**

Макалада булут технологияларынын санарип маалыматтарга, Ардуинонун аппараттык-программалык чөйрөсүн колдонуу менен маалымат чогултуу, иштетүү жана аларды тармак аркылуу булутка жиберүүнү пайдалануунун чоң мүмкүнчүлүктөрү, пайдалануу концепциялары каралган. Макалада булут технологияларынын актуалдуулугу, аны колдонуудагы артыкчылыктар жана AWS (Amazon Web Services) те сунушталган бекер программдык продукттар ачык көрсөтүлгөн.

Ачык сөздөр: булут технологиялары; ардуино; маалымат технологиялары.

В представленной работе рассматриваются сильные стороны и возможности применения концепции облачных технологий и использования аппаратно-программной среды arduino для построения простых систем автоматизации для сбора и обработки данных. В статье освещаются актуальность и преимущества облачных вычислений и бесплатных программных продуктов, предлагаемых на примере AWS (Amazon Web Services).

Ключевые слова: облачные технологии; ардуино; информационные технологии.

The present paper considers the strengths and potential of the concept of cloud technologies for a variety of digital data. Creating an environment based on Arduino for collecting and processing data from the sensors and output data over the network in a cloud. The article highlights the relevance and benefits of using cloud technology and free software products offered on AWS Example (Amazon Web Services).

Keywords: informational technologies; cloud computing; arduino.

Облачные технологии относятся к приложениям и цифровым услугам, которые находятся исключительно в интернете и расположены на серверных системах. Данные в облаке «централизованно размещаются, управляются или хранятся в государственных или частных хранилищах, что сегодня мы называем «облачными услугами». Эти хранилища также известны как центры обработки данных. Благодаря массовому производству интеллектуальных датчиков и широкому распространению интернета, мы можем использовать их в самых различных сферах.

Одним из актуальных сфер применения облачных вычислений используется в области окружающей среды и изменения климата. С точки зрения окружающей среды и изменения климата включают следующее: в экологическом секторе «облако» позволяет совместное использование больших цифровых массивов данных посредством таких инициатив, как Глобальная система систем наблюдения Земли (GEOSS, Global Earth Observation System of Systems). Одним из примеров является совместное использование данных об окружающей среде, таких как данные о биологических связях по всему миру. Они также позволяют совместное пользование мощными компьютерами для анализа экологических данных, записанных с различных экологических сенсорных платформ. Распределенные (грид) вычисления являются вариацией облачных вычислений, где компьютеры способствуют вычислительной мощности для решения сложных расчетов и/или проведения моделирования сложных явлений, таких как погодные условия, мониторинг сейсмического ускорения, нагрузки, температуры и другие экологические или экосистемные явления, события или

процессы. Благодаря широкополосному соединению, исследователи и стратегические аналитики во всем мире могут получить доступ к этим грид услугам для проведения исследований и анализа своих собственных наборов данных. Облачные решения как SaaS («программное обеспечение как услуга») PaaS («платформа как услуга») и IaaS («инфраструктура как услуга») становятся движущей силой трансформаций и превращают ИТ-инфраструктуру из центра затрат в источник инноваций, роста и неиссякаемых возможностей эффективности.

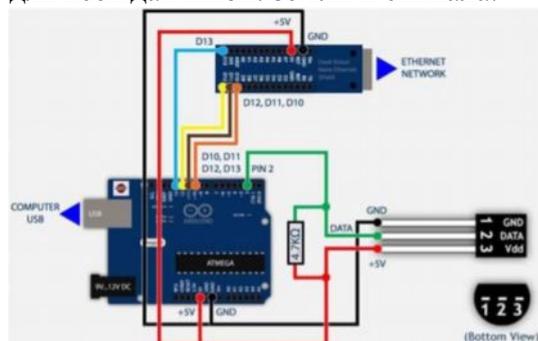
Описание аппаратно-программной платформы Arduino. Arduino – это простая в использовании открытая электронная платформа, включающая программное обеспечение и предназначенная для быстрого создания интерактивных электронных устройств. Она была создана группой энтузиастов, которые позиционировали свою разработку как платформу для быстрой реализации небольших проектов. Arduino строится на базе микроконтроллеров Atmel и используется для получения сигналов от аналоговых и цифровых датчиков, управления различными исполнительными устройствами и обмена информацией с компьютером при помощи различных интерфейсов. Сенсоры и датчики подключаемые к плате Ардуино функционируют для сбора и передачи данных. При получении данных о внешней среде, устройство обрабатывает полученный сигнал и отправляет данные в центральное хранилище данных или сервер где эти данные анализируются и хранятся для дальнейшего исследования. Измерительных приборов на сегодняшний день много: сенсоры, простые датчики температуры, приборы учета потребления и т.д, они обладают четырьмя основными возможностями: зондирование, обработка данных, передача данных и, в некоторых случаях, активирование, то есть управление иными устройствами, такими как электродвигатели, другие датчики, системы сигнализации и т.д., такие преобразователи могут обнаруживать энергию различных физических, химических и других явлений и преобразовывать ее в сигнал, электрический импульс и т.д. Сенсорные технологии применялись и применяются во многих областях такие как национальная безопасность, наземный мониторинг как поверхности земли, так и воды, сбор информации в целях защиты, мониторинг окружающей среды, анализ и прогнозирование погоды и климата, наблюдение и мониторинг зоны конфликта, мониторинг сейсмического ускорения, нагрузки, температуры, скорости ветра и GPS-данных. Ардуино дает возможность достичь высокого уровня автономности, намного упрощает процесс работы с микроконтроллерами и имеет ряд преимуществ перед другими устройствами. Первое это низкая стоимость платы относительно дешево по сравнению с другими платформами. Самая недорогая версия модуля Arduino может быть собрана в ручную, а некоторые даже готовые модули стоят меньше 50 долларов. Кросс-платформенность – программное обеспечение Arduino работает под ОС Windows, Macintosh OSX и Linux. Большинство микроконтроллеров ограничивается ОС Windows. Простая и понятная среда программирования – среда Arduino подходит как для начинающих пользователей, так и для опытных. Arduino основана на среде программирования Processing, что очень удобно для начинающих программистов, работающие с данной средой. Программное обеспечение с возможностью расширения и открытым исходным текстом так же может дополняться библиотеками C++. Пользователи, желающие понять технические нюансы, имеют возможность перейти на язык AVR C на котором основан C++. Соответственно, имеется возможность добавить код из среды AVR-C в программу Arduino.

Разработка автоматизированной системы с использованием облачных технологий. Для разработки среды в качестве аппаратных средств были выбраны :**Arduino UNO** - центральный контроллер, недорогой микропроцессор с открытой архитектурой, способный считывать входные данные в виде напряжения на своих аналоговых контактах. Если к определенным входам устройства подключить датчики, то оно будет программным способом считывать информацию с этих контактов.

Ethernet Shield ENC28J60 - сетевой модуль предназначен для реализации сетевых приложений: удаленного управления системой, сетевого доступа, публикации данных.

Максимальная скорость передачи данных 100 Мбит/с. с разъемом RJ45 который крепится к плате с помощью специальных пластиковых штырей-фиксаторов.

Температурный датчик DS18B20 - этот датчик температуры основан на популярной микросхеме DS18B20. Он позволяет определить температуру окружающей среды в диапазоне от -55°C до $+125^{\circ}\text{C}$ и получать данные в виде цифрового сигнала с 12-битным разрешением по 1-Wire протоколу. Этот протокол позволит подключить огромное количество таких датчиков, используя всего 1 цифровой порт контроллера, и всего 2 провода для всех датчиков: земли и сигнала.



Программирование сборки ведется целиком через собственную программную оболочку Arduino (IDE), бесплатно доступную на сайте arduino.cc. В этой оболочке имеется текстовый редактор, менеджер проектов, препроцессор, компилятор и инструменты для загрузки программы в микроконтроллер. Оболочка написана на Java на основе проекта Processing, работает под Windows, Mac OS X и Linux. Программы, написанные программистом Arduino называются наброски (или иногда скетчи - варваризм от англ. sketch) и сохраняются в файлах с расширением `.ino`. Эти файлы перед компиляцией обрабатываются препроцессором Ардуино. Закачка программы в микроконтроллер Ардуино происходит через предварительно запрограммированный специальный загрузчик (все микроконтроллеры от Ардуино продаются с этим загрузчиком). Загрузчик создан на основе Atmel AVR Application Note AN109. Загрузчик может работать через интерфейсы RS-232, USB или Ethernet в зависимости от состава периферии конкретной процессорной платы. Для использования шилдов или датчиков в среде разработки ардуино нам необходимо установить библиотеки. Библиотеки рекомендуются скачивать с сайта Github, так как именно там можно скачать последние версии. Для интернет шилда (ENC28J60) была использована библиотека "UIPEthernet", что же касается температурного датчика DS18B20 то здесь необходима библиотека "OneWire" которую можно скачать с сайта Github.

Для проектирования проекта на платформе ардуино была использована облачная технология **Amazon Elastic Compute Cloud (EC2)** – это веб-сервис, который используется для создания и запуска виртуальных машин в облаке. Такие виртуальные машины называются инстансами. Время запуска одного инстанса обычно не превышает двух минут. Сразу после запуска инстанс начинает работать, к нему есть root-доступ по SSH или Remote Desktop. Средствами EC2 со встроенным образом операционной системы был развернут Ubuntu с установкой и настройкой веб сервера Apache, PHP и mysql. Для дальнейшего взаимодействия платы ардуино и веб сервера необходимо написать скрипты на php которые включают в себя три этапа

- 1). Соединение с базой (*таблица.2.*)
- 2). Получение данных (*таблица.3.*)
- 3). Вывод на веб странице полученные данные (*таблица.4 .*)

Для самой базы данных в mysql создаем БД *test* с таблицей *temperature* (*таблица.1.*)

На завершительном этапе в браузере должно получиться как на (*рис.1.*)

Таблица 1 - Создание базы данных test

```
CREATE TABLE `test`.`temperature` ( `id` INT NOT NULL AUTO_INCREMENT
PRIMARY KEY COMMENT 'unique ID', `event` TIMESTAMP NOT NULL DEFAULT
CURRENT_TIMESTAMP COMMENT 'Event Date and Time', `sensor` VARCHAR( 30 ) NOT
NULL COMMENT 'Unique ID of the sensor', `celsius` VARCHAR( 10 ) NOT NULL COMMENT
'Measured Temperature in Celsius', INDEX ( `event`, `sensor` )) ENGINE = InnoDB;
```

Таблица 2 - Подключение к базе connect.php

Таблица 3 - Получение данных add_data.php

```
1. <?php function Connection() {
2.     $server="localhost";
3.     $user="****";
4.     $pass="****";
5.     $db="test";
6.     $connection = mysql_connect($server,
7.     $user, $pass);
8.     if (!$connection) {
9.         die('MySQL ERROR: ' . mysql_error());
10.    }
11.    mysql_select_db($db) or die( 'MySQL
12.    ERROR: ' . mysql_error() );
    return $connection;
} ?>
```

```
1. <?php include("connect.php");
2.     $link=Connection();
3.     $temp1=$_POST["temp1"];
4.     $query = "INSERT INTO `tempLog`
5.     (`temperature`, `humidity`)
6.     VALUES (". $temp1 .")";
7.     mysql_query($query,$link);
8.     mysql_close($link);
9.     header("Location: index.php");
10.    ?>
```

```
1. <?php
```

Таблица 4 - Вывод данных на веб странице index.php

```
2. include("connect.php");
3. $link=Connection();
4. $result=mysql_query("SELECT * FROM19.
5. `tempLog` ORDER BY `timeStamp`
6. DESC",$link);
7. ?>
8. <html>
9. <head>
10. <title>Sensor Data</title>
11. </head>
12. <body>
13. <h1>Temperature </h1>
14. <table border="1" cellspacing="1"5.
15. cellpadding="1">
16. <tr>
17. <td>&nbsp;Date&nbsp;</td>
18. <td>&nbsp;Temperature&nbsp;</td>
```

```
16. </tr>
17. <?php
18. if($result!==FALSE){
19.     while($row =
20.     mysql_fetch_array($result)) {
21.         printf("<tr><td> &nbsp;%s </td><td>
22.         &nbsp;%s&nbsp;</td><td>
23.         &nbsp;%s&nbsp;</td></tr>",
24.         $row["timeStamp"],
25.         $row["temperature"]);
26.     }
27.     mysql_free_result($result);
28.     mysql_close();
29. }
30. ?>
31. </table>
32. </body>
33. </html>
```

ID	Date and Time	Sensor Serial	Temperature in Celsius
7	2014-03-26 11:02:38	288884820500006X	12.3
8	2014-03-26 11:02:44	288884820500006X	12.3
9	2014-03-26 11:02:49	288884820500006X	12.3
10	2014-03-26 11:02:55	288884820500006X	12.3

Рис.1. Тестирование Базы данных test

Таким образом был создан веб сервер с необходимыми скриптами на php и mysql таблицей осталось написать ардуино код (таблица 5) для получения температурных данных датчика **DS18B20**.

Таблица 5 - Подключение температурного датчика и Ethernet шилда

1.	#include <UIPEthernet.h>	16.	{ Ethernet.begin(mac, ip);
2.	#include <OneWire.h>	17.	sensors.begin();
3.	#include <DallasTemperature.h>	18.	Serial.begin(9600);
4.	##include <Etherne#include <SPI.h>	19.	pinMode (pirPin, INPUT);}
5.	byte mac[] = { 0x90, 0xA2, 0xDA, 0x20, 0x4C, 0xFB };	20.	void loop()
6.	byte ip[] = { x, x, x, x }; ip ардуино	21.	{
7.	byte server[] = { x, x, x, x }; //ip адрес сервера (aws)	22.	sensors.requestTemperatures();//получаем данные сенсора
8.	EthernetClient client;	23.	delay(200);
9.	#define DHTPIN A0 //Контакт к которому подключен датчик температуры и влажности	24.	int t = sensors.getTempCByIndex(0);
10.	#define ONE_WIRE_BUS 7 // Выбираем сенсор который DHT11	25.	String msg = "GET /« ip адрес сервера AWS» ?t="+ String(t);
11.	#define DHTTYPE DHT11	26.	Serial.println(msg);
12.	OneWire oneWire(ONE_WIRE_BUS);	27.	client.connect(server, 80);
13.	DallasTemperature sensors(&oneWire);	28.	client.println(msg);
14.	int pirPin = 2; //digital 2	29.	delay (10000);
15.	void setup()	30.	client.stop();
		31.	//Данные будут отправляться каждые 10 сек, можно изменить delay (8000); }

Заключение. Таким образом, рассмотренный пример метеостанции, разработанный на базе ардуино и использования облачных вычислений показанный на схеме (Рис.2) дает возможность форматировать и архивировать метеоданные с индикаторами (время сбора, температуры воздуха, и т.д.) для более эффективного прогнозирования на региональном и локальном уровнях.



Рис.2. Взаимодействие ардуино и облачной технологии

Список литературы

1. James Harris and Steven Nunn, "Cloud Computing's Great Promise", 30 June 2010. Доступно на странице <http://www.forbes.com/2010/06/30/pharmaceuticals-mobile-salesforcecom-technology-cloud-computing.html?boxes=Homepagetoprated>.
2. Geos, "GEOSS on-line", Доступно на странице <http://www.earthobservations.org/>.
3. Википедия, "Herbarium", Доступно на странице <http://en.wikipedia.org/wiki/Herbarium>.
3. Arduino Ethernet - Pushing data to a (PHP) server Доступно на странице <http://www.tweaking4all.com/hardware/arduino/arduino-ethernet-data-push>