

Омский Научный центр Сибирского отделения Российской академии наук
Региональная общественная организация «Омский совет ректоров»
Омское региональное отделение Всероссийской общественной организации
«Русское географическое общество»
Детская областная общественная организация
«Научное общество учащихся «Поиск»
Бюджетное общеобразовательное учреждение Полтавского муниципального района
«Полтавский лицей»

ЛП

Межрегиональная научно-практическая конференция школьников и учащейся молодежи

Тема: «Исследование возможности реализации системы
автоматической подачи школьных звонков на базе ESP8266»
Учебно-исследовательская работа
Научное направление: информатика 5-9 классы

Выполнил:
ученик 9 класса
БОУ «Полтавский лицей»
Кондратенко Данила Александрович

Научный руководитель:
ШМИТ
Савченко Сергей Викторович

Омск — 2020

Оглавление

Введение.....	3
Определения.....	4
Описание реализации.....	5
Структура системы автоматической подачи звонков.....	5
Исполнительный блок системы.....	5
Протокол взаимодействия.....	7
Панель управления.....	9
Пользовательский интерфейс.....	9
Хранение расписания на компьютере.....	10
Заключение.....	10
Список использованной литературы.....	11

Введение

В современном мире происходит автоматизация рутинных процессов, из-за чего труд людей может быть облегчен путём возложения части работы на компьютер. Так можно облегчить работу школьного вахтёра, в частности, автоматизировать подачу школьных звонков. Это исключит человеческий фактор, так как вахтёр может ненадолго отлучиться от своего рабочего места, а в это время должен начаться урок или перемена.

Цель работы: реализовать систему автоматической подачи звонков.

Задачи:

1. изучить язык C++ и его стандартную библиотеку шаблонов;
2. изучить библиотеки ESP8266;
3. создать сервер автоматической подачи звонков на основе ESP8266;
4. изучить фреймворк Qt;
5. реализовать панель управления системой автоматической подачи звонков при помощи C++/Qt.

Методы исследования:

- изучение справочной литературы (документация Qt, RFC);
- попытка создания продукта, предусмотренного темой исследования.

Определения

ESP8266 – микроконтроллер, разработанный в китайской компании Espressif. Оснащён 32-битным процессором Tensilica Xtensa L106 с тактовой частотой 80 МГц. Поддерживает Wi-Fi, что во многом определяет его применение.

Фреймворк (англ. *framework* – каркас) — набор заготовок и шаблонов, определяющий архитектуру программного обеспечения.

Qt – кроссплатформенный *фреймворк* для разработки графических приложений на C++. Имеет привязки к иным языкам программирования, например, привязки к Python 3 – PyQt и её ответвление PySide.

Отличается использованием метаобъектного компилятора, который занимается предварительной обработкой кода.

Начал своё развитие в 1995 году. Начиная с версии 4.5, распространяется по трём лицензиям: Qt Commercial, GNU GPL¹ и GNU LGPL² (для разнолицензионных компонентов).

C++ – компилируемый язык программирования, разработанный Бьярне Струострупом на основе языка Си в 1983 году. Типизация статическая, т. е. объявленная в коде переменная не способна менять свой тип в ходе работы программы. Большое внимание в языке уделяется развитию объектно-ориентированного программирования, т. е. представлению программы в виде совокупности работающих объектов, относящихся к разным классам.

Стандартизован Международной организацией по стандартизации (ISO). Последний стандарт имеет название C++17.

1 GNU General Public License – Универсальная общественная лицензия GNU. Лицензию на свободное программное обеспечение, в котором весь используемый код лицензирован только под этой лицензией.
2 GNU Lesser General Public License – Менее универсальная общественная лицензия GNU. Лицензия на свободное программное обеспечение, позволяющая компоновать библиотеки под другими лицензиями.

Описание реализации

Структура системы автоматической подачи звонков

Система автоматической подачи звонков состоит из двух частей: исполнительного блока (ESP8266) и программного обеспечения на компьютере пользователя.

Взаимодействие компьютера и исполнительного блока осуществляется посредством беспроводной технологии Wi-Fi. Главное преимущество использования этого способа взаимодействия — удобство монтажа, которое достигается за счёт излучения радиоволн, а значит, и отсутствия проводов.

Для управления компьютер должен быть подключён к исполнительному блоку либо напрямую, т. е. при помощи Wi-Fi-адаптера, встроенного в компьютер или подключенного при помощи USB, либо посредством Wi-Fi-приёмника, включённого в локальную вычислительную сеть школы или кабинета информатики.

Исполнительный блок системы

Исполнительный блок системы автоматической подачи звонков выполнен на микроконтроллере ESP8266. В его оперативной памяти сохраняется расписание уроков в виде массива 16-битных беззнаковых целых чисел *lessons* и массива 8-битных беззнаковых целых чисел *rings*. Временной интервал между уроками задаётся парой целых чисел от 0 до 1440, каждое из которых показывает время, прошедшее с полуночи, в минутах. Промежуток $[lessons_{2n}; lessons_{2n+1})$ задаёт временные рамки урока n . Промежуток $[lessons_{2n-1}; lessons_{2n})$ задаёт интервал между уроками $n-1$ и n , т. е. перемену.

В программе исполнительного блока предусмотрен внутренний счёт времени.

Алгоритм работы исполнительного блока таков:

1. Обрабатываются HTTP-запросы³, отправляемые с компьютера.

3 HTTP – Hypertext Transmission Protocol (протокол передачи гипертекста). Основной протокол, используемый для передачи файлов и текстовых данных по каналам Интернета. Используется на веб-сайтах.

2. Берётся значение прошедших с начала дня минут M и проверяется на совпадение со значениями в массиве $lessons$. Здесь i – номер урока.
- Если M совпадает с временем начала i -го урока либо меньше времени окончания i -го урока ($lessons_{2i} \leq M < lessons_{2i+1}$), то подать звонок на урок: 2 раза по 3 секунды с паузой между ними в 5 секунд.
 - Если M совпадает с временем окончания i -го урока либо меньше времени начала $(i+1)$ -го урока ($lessons_{2i+1} \leq M < lessons_{2i+2}$), то подать звонок на перемену: один звонок продолжительностью 3 секунды, далее пауза в одну секунду и $rings_i$ звонков продолжительностью в одну секунду с равным промежутком между звонками. Если $rings_i = 0$, то после длинного звонка ничего не делать.

Таким образом, исполнительный блок одновременно работает как сервер HTTP и как управляющий реле звонка.

Протокол взаимодействия

Чтобы система автоматической подачи звонков могла получать запросы на изменение расписания или времени был разработан протокол взаимодействия, основанный на передаче формы в MIME-формате⁴ `application/x-www-form-urlencoded` и использующий для транспортировки протокол HTTP.

Кодирование формы в формате URL⁵ происходит следующим образом. Берутся пары вида «ключ-значение», состоящие из двух строк. В них зарезервированные символы кодируются в виде %XX, где XX – шестнадцатеричное число. Символ процента, являющийся индикатором закодированного символа, кодируется как %25 в первую очередь. Пробел может заменяться на знак '+」.

%21	%22	%23	%24	%26	%27	%28	%29	%2A	%2B	%2C	%2F	%3A	%3B	%3D	%3F	%40	%5B	%5D
!	"	#	\$	&	'	(	)	*	+	,	/	:	;	=	?	@	[	]

Таблица 1. Зарезервированные символы в URL

Незарезервированные символы, а также цифры и латинские буквы не должны кодироваться таким образом.

-	—	.	!
---	---	---	---

Таблица 2. Незарезервированные символы в URL

Ключ и значение в парах соединяются знаком равенства, а сами пары — амперсандами.

Исходные данные:

Ключ	Значение
to	philosoph@example.net
message	Good afternoon.

Результат:

to=philosoph%40example.net&message=Good+afternoon.

Таблица 3. Пример URL-кодирования формы

-
- 4 MIME – Multipurpose Internet Mail Extensions (многоцелевые расширения интернет-почты). Стандарт, описывающий передачу различной информации внутри текстовых данных, в т.ч. по электронной почте. Является неотъемлемой частью HTTP.
- 5 URL – Universal Resource Locator (универсальный указатель на ресурс).

Для декодирования формы строка формы делится на пары «ключ-значение» по амперсандам, пары разделяются по знаку равенства, в строках из пар заменяются все закодированные символы.

Каждый запрос представляет собой форму, состоящую из:

- поля метода;
- полей аргументов (если они принимаются).

В запросе могут применяться следующие методы:

Метод	Действие
set	Запрос изменения расписания или установки времени
schedule	Запрос текущего установленного расписания
doring	Команда на подачу звонка или серии звонков

Таблица 4. Методы системы автоматической подачи звонков

Методы set и doring используют поля аргументов.

Наименование	Значение
lessnum	Количество уроков в расписании
schedule	Само расписание
time	Время на часах системы

Таблица 5. Аргументы метода set

Наименование	Значение
number	Количество подаваемых звонков
time	Время одного звонка
pause	Пауза между звонками (необязательно, если кол-во звонков меньше двух)

Таблица 6. Аргументы метода doring

В ответ система возвращает URL-закодированную форму, но уже в MIME-формате text/plain. Для методов set и doring ответом будет state=0. Для метода schedule ответом будет форма, состоящая из двух полей: lessnum (количество уроков) и schedule (собственно расписание). Каждый урок представлен в виде N-S.E.R, где N – номер урока, S – время начала урока в минутах, E – время окончания урока в минутах, R – количество звонков после звонка на переменную. Записи разделяются символом '_'.

Панель управления

Пользовательский интерфейс

Интерфейс программы для управления системой автоподачи звонков по локальной сети должен быть понятен конечному пользователю, поэтому он был реализован подобно интерфейсу Microsoft Office Excel или LibreOffice Calc. Для создания такого графического интерфейса понадобились, в частности, классы `QStandardItemModel` и `QTableView` из Qt.

Таблица с расписанием уроков занимает большую часть окна. Редактировать его можно как обычную таблицу (режим простого редактирования) или с последующим пересчётом (режим калькулятора). В режиме простого редактирования расписание представлено в виде таблицы, состоящей из трёх столбцов: начало урока, конец урока и количество звонков после звонка на перемену, а в режиме калькулятора — как таблица из четырёх столбцов: начало урока, длина урока, длина перемены и количество звонков после звонка на перемену. Любая временная отметка в расписании представлена в виде ЧЧ:ММ, где ЧЧ — две цифры часов, ММ — две цифры минут.

В этой программе предусмотрена функция расчёта нового расписания по количеству уроков, их длительности, времени начала первого урока и длительности перемен. Рассчитанное расписание передаётся в основную программу при помощи предусмотренной в Qt системы обработки событий, основанной на так называемых сигналах и слотах. Сигналы являются функциями, реализуемыми при помощи метаобъектного компилятора Qt и вызывающими привязанную функцию — «слот», которую реализует программист.

Также предусмотрена возможность проверки расписания, составленного вручную. Правильным расписанием в данном случае будет считаться такое расписание, в котором уроки расположены в порядке возрастания начала урока, а конец урока должен быть позже начала.

Хранение расписания на компьютере

Панель управления системой автоматической подачи звонков сохраняет расписание в виде файла в формате `.shdl`, который обрабатывается как CSV-таблица.⁶ Каждая строка состоит из трёх значений, разделённых запятыми: номер урока, время начала урока и время окончания урока. Временные отметки представлены в виде ЧЧ:ММ, где ЧЧ — две цифры часов, ММ — две цифры минут.

Заключение

В результате исследования я пришёл к выводу, что система автоматической подачи звонков на ESP8266 и программное обеспечение для компьютера поддаётся реализации. Кроме того, средства, использовавшиеся при создании системы автоматической подачи звонков и программного обеспечения для компьютера, применимы для создания более сложных программ.

Таким образом, можно считать, что цель, поставленная в начале работы, достигнута.

6 CSV – Comma Separated Values (значения, разделённые запятыми).

Список использованной литературы

1. Qt // Википедия [Электронный ресурс]. - Режим доступа:
<https://ru.wikipedia.org/wiki/Qt>
2. C++ // Википедия [Электронный ресурс]. - Режим доступа:
<https://ru.wikipedia.org/wiki/C++>
3. Tim Berners-Lee et al. RFC 3986: Uniform Resource Identifier (URI): Generic Syntax. [Electronic resource]. - URL: <https://tools.ietf.org/html/rfc3986>
4. Tim Berners-Lee et al. RFC 1866: Hypertext Markup Language — 2.0. §8.2.1. The form-urlencoded Media Type. [Electronic resource]. - URL:
<https://tools.ietf.org/html/rfc1866>
5. Signals and Slots // Qt 5 documentation. [Electronic resource]. - URL:
<https://doc.qt.io/qt-5/signalsandslots.html>