Комплект разработки

программы Zvonok

Содержание

[1. Аннотация 1](#_Toc111806449)

[2. Состав комплекта 1](#_Toc111806450)

[3. Настройка 1](#_Toc111806451)

[4. Установка на ПК 2](#_Toc111806452)

[5. Описание работы 2](#_Toc111806453)

# Аннотация

Программа предназначена для управления подачей звуковых и аудио сигналов (сообщений) в соответствии с задаваемым расписанием.

Программа работает под ОС Microsoft Windows 7, 10.

# Состав комплекта

* 1. distributive – дистрибутив.
* Borland C++ Builder Enterprise 6.0;
* SetupVCD5500;
  1. documents – документация;
* Описание Zvonok.docx;
  1. CBsource – исходный код программы для Borland C++ Builder.

# Настройка

* 1. Разработка производилась под ОС Microsoft Windows 7 Service Pack 1;
  2. Установите на ПК эмулятор CDSetupVCD5500.
  3. Установите BorlandC++ Builder подключая образа CDдисков через эмулятор CD.

- подключите образ диска Borland C++ Builder 6.0 Enterprise [cd1].iso.

- запустите на виртуальном CD программу install.exe.

- в программе установки выберите пакет«C++ Builder 6».

- при запросе программы установки «SerialNumber», запустите программу Borland C++ Builder 6.0 keygenerator.exe,

- в программе Borland Keygen нажмите кнопку generate.

- скопируйте в программу установки сгенерированные коды.

- далее следуйте программе установки,

- по запросу программы установки замените на виртуальном CD образ диска Borland C++ Builder 6.0 Enterprise [cd2].iso.

* 1. При первом запуске C++ Builder выберите вариант регистрации «Register or activate by phone…»
* запустите программу Borland C++ Builder 6.0 keygenerator.exe,
* скопируйте коды из программы Borland C++ Builder в программу Borland Keygen, получите ActivationKey и укажите его в программе C++ Builder.
  1. ВC++ Builder открыть проект Zvonok.bpr из каталога CBsource.

# Установка на ПК

* 1. Скопируйте каталог Release и запустите файл Zvonok.exe

# Описание работы

* 1. Настройка
     1. При запуске программа считывает настройки из файла setup.ini. При его отсутствии - из файла старого формата par.txt. Если нет никаких файлов, то параметры настройки устанавливаются по умолчанию.

Настройки хранятся в структуре set\_setup.

* + 1. При запуске, в методе FormCreate программа устанавливает значение задержки включения усилителя звукового сигнала в поле структуры

set\_setup.up\_booster\_delay = DEF\_UP\_BOOSTER\_DELAY

, значение 3 по умолчанию.

В функции OpenFiles программа считывает значение задержки из файла. Если при открытии файла, его чтении или сохранении диапазона значений (1..99) возникают ошибки, программа оставляет значение по умолчанию.

* + 1. При запуске, в методе FormCreate программа устанавливает значение задержки выключения усилителя звукового сигнала в поле структуры

set\_setup.down\_booster\_delay = DEF\_DOWN\_BOOSTER\_DELAY

, значение 1 по умолчанию.

В функции OpenFiles программа считывает значение задержки из файла. Если при открытии файла, его чтении или сохранении диапазона значений (1..99) возникают ошибки, программа оставляет значение по умолчанию.

* + 1. При запуске, в методе FormCreate программа устанавливает значение часового пояса в поле структуры

set\_setup.time\_zone = DEF\_TIME\_ZONE

, значение 3 по умолчанию.

В функции OpenFiles программа считывает значение часового пояса из файла. Если при открытии файла, его чтении или сохранении диапазона значений (2.12) возникают ошибки, программа оставляет значение по умолчанию.

* + 1. При запуске, в методе FormCreate программа устанавливает значение флага перехода на зимнее время в поле структуры

set\_setup.winter\_time = DEF\_WINTER\_TIME

, значение 0 по умолчанию.

В функции OpenFiles программа считывает значение флага из файла. Если при открытии файла, его чтении или сохранении диапазона значений (0..1) возникают ошибки, программа оставляет значение по умолчанию.

* + 1. При запуске, в методе FormCreate программа устанавливает имя звукового файла для внешнего сигнала «Тревога» в поле структуры

set\_setup.alarm\_audio\_file = DEF\_ALARM\_SOUND

, значение “Terror.wav” по умолчанию.

Программа проверяет наличие этого файла в директории программы и устанавливает флаг его наличия

set\_setup.are\_sound\_files += IS\_DEF\_ALARM

В функции OpenFiles программа считывает имя звукового файла “Тревога” из файла настроек. Если при открытии файла настрок, его чтении или поиске звукового файла возникают ошибки, программа оставляет значение по умолчанию.

* + 1. При запуске, в методе FormCreate программа устанавливает имя звукового файла для внешнего сигнала «Пожар» в поле структуры

set\_setup.fire\_audio\_file = DEF\_FIRE\_SOUND

, значение “Fire.wav” по умолчанию.

Программа проверяет наличие этого файла в директории программы и устанавливает флаг его наличия

set\_setup.are\_sound\_files += IS\_DEF\_FIRE

В функции OpenFiles программа считывает имя звукового файла “Пожар” из файла настроек. Если при открытии файла настроек, его чтении или поиске звукового файла возникают ошибки, программа оставляет значение по умолчанию.

* + 1. При запуске, в методе FormCreate программа устанавливает имя звукового файла для тестового звукового сигнала в поле структуры

set\_setup.test\_audio\_file = DEF\_TEST\_SOUND

, значение “Ring.wav” по умолчанию.

Программа проверяет наличие этого файла в директории программы и устанавливает флаг его наличия

set\_setup.are\_sound\_files += IS\_DEF\_TEST\_SOUND

В функции OpenFiles программа считывает имя тестового звукового файла из файла настроек. Если при открытии файла настроек, его чтении или поиске звукового файла возникают ошибки, программа оставляет значение по умолчанию.

* + 1. Если при запуске программа не обнаружила звуковых файлов, указанных в файле настроек или не обнаружила звуковых файлов заданных по умолчанию, то программа отображает окно с выводом ошибок.
    2. Программа считывает расписание событий из файла Schedule.sch.

Формат файла:

#

<параметры события>

<имя файла звукового сообщения>

<строка 1 описания события>

…

<строка n описания события>

#

, где символ # - определяет начало новой записи.

Параметры события записываются в виде строки:

<year> <month> <day> <weekday> <use\_year> <use\_month> <use\_day> <use\_weekly> <ex\_day>

<use\_hour> <use\_minute> <use\_second> <hour> <minute> <second>

<cycle\_flags> <event\_sign> <row\_count>

, где параметры разделены пробелом, а количество символов каждого параметра произвольное,

параметры описаны в структуре t\_Event,

row\_count – количество строк описания события (n)

При отсутствии файла расписания программа может считать расписание из файлов старого формата: day.txt, rxx.txt (где хх – от 00 до 99), ex.txt

* + 1. Если при запуске программа не обнаружила файлов расписания, в окне программы будет отображаться сообщение «Добавьте событие…»
  1. Формирование расписания
     1. Все события считанные из файла сохраняются в векторе BaseEvent.

Расписание должно иметь

- наименование (понедельник – воскресенье, будни);

- список событий (дата, время, звуковой файл, флаг звонка, текст, флаг повтора).

* 1. Обработка событий расписания
     1. Программа должна контролировать наступление события по:

- минуте;

- часу;

- дню недели;

- дате (день, месяц, год).

* + 1. Программа, по наступлению события должна выдать сигнал
    2. Программа не должна обрабатывать пропущенные события при перезапуске.
  1. Ручное управление
     1. Программа должна считывать состояние кнопок управления:

- кнопка «\*» - выбор режима;

- кнопка «<<» - переход назад;

- кнопка «>>» - переход вперед;

- кнопка «Гр» - регулирование громкости;

- кнопка «>» - кнопка воспроизведения.

* + 1. При нажатии кнопки «\*» программа должна устанавливать режим в цикле: «Расписание»->«Состояние» -> «Часовой пояс»
    2. При нажатии кнопки «>» программа должна:

- выдать сигнал без звонка, если нет режима редактирования,

- установить выбранное пользователем значение в режиме редактирования.

* + 1. При нажатии кнопки «Гр» программа должна устанавливать режим в цикле: «Громкость микрофона» -> «Громкость сообщений» -> «Шрифт индикатора».
    2. При нажатии кнопок «<<» , «>>» программа должна:

- в режиме «Расписание» отображать список расписаний в формате:

\*<название расписания> -><hh:mm>, где

«\*» - если отображается текущее расписание,

«Пон» ..«Вск» - расписание на дни недели,

<hh:mm> - время следующего события,

«--:--» отображается в поле времени,если все события дня закончены;

- в режиме «Состояние» отображать список ошибок или «ОК»;

- в режиме «Часовой пояс» - изменять значение в диапазоне: 2..12

- в режиме «Громкость микрофона» изменять величину громкости микрофона в диапазоне…;

- в режиме «Громкость сообщений» изменять величину громкости сообщений в диапазоне…;

- в режиме «Шрифт индикатора» изменять значения в диапазоне: «шрифт1».. «шрифт 4».

* + 1. Программа должна отображать состояние управления на индикаторе.
  1. Выдача сигналов
     1. Программа должна выдавать сигнал в следующем порядке

- подать сигнал на линию звонка, если указано,

- подать сигнал на включение усилителя звукового сигнала,

- воспроизвести звуковой файл (mp3),

- после проигрывания файла отключить усилитель с заданной задержкой.

* + 1. Программа должна понижать уровень громкости для входящих аудио во время выдачи сигнала.
  1. Программа должна контролировать сигнал от микрофона
     1. При подключении микрофона программа должна отключать усилитель звукового сигнала, если производится выдача сигнала.
  2. Получение даты и времени по каналу GPS
     1. Программа должна принимать данные с GPS и синхронизировать системное время.
     2. Программа должна выводить на индикатор текущие дату и время в формате:<dw>, <dd><mnth><hh>:<mm>.
     3. При отсутствии / недостоверности данных с GPS программа должна отображать на индикаторе сообщение «Нет данных GPS»
     4. При наличии ошибок GPSпрограмма должна отображать на индикаторе сообщение «GPSне исправен».
  3. Обработка линейных сигналов
     1. Программа должна отслеживать состояние линий:

- «пожар»,

- «охрана»,

- дублер кнопки «>».

* + 1. Программа должна выдать сигнал без звонка при

- срабатывании сигнала на линии «Пожар»,

- срабатывании сигнала «охрана»,

- срабатывании сигнала «>».

# Отладка

1. Функция malloc работает не стабильно. При ее использовании при закрытии программы появляется ошибка при попытке освободить память – лучше не использовать.
2. Для элемента CheckBox при ручном изменении параметра Checked автоматически вызывается функция Click().