Содержание

Лабораторная работа №1

Цель работы: ознакомление с интерфейсом программы BricxCC, освоение навыков вывода информации на дисплей, использования таймера и воспроизведения звуков через встроенный динамик.

В этой и последующих лабораторных работах для программирования NXT Brick на языке NXC будет использоваться программа BricxCC. Подключение NXT Brick к компьютеру начинается с запуска программы BricxCC, при запуске которой появляется меню наподобие Рис.1.1.

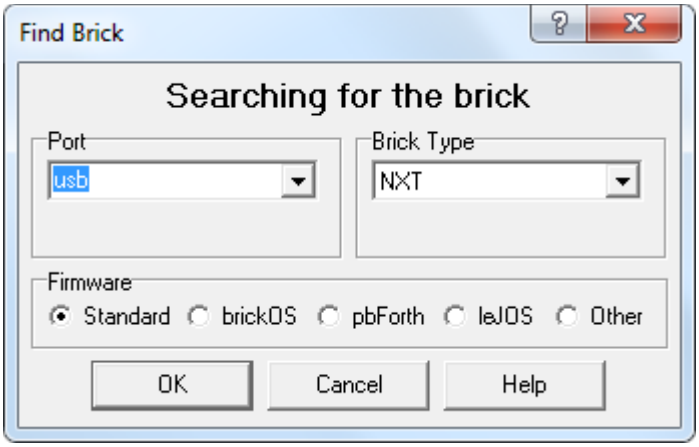

Рис.1.1. Меню поиска подключённого блока NXT

После подключения блока NXT появляется интерфейс программы, наподобие Рис.1.2.

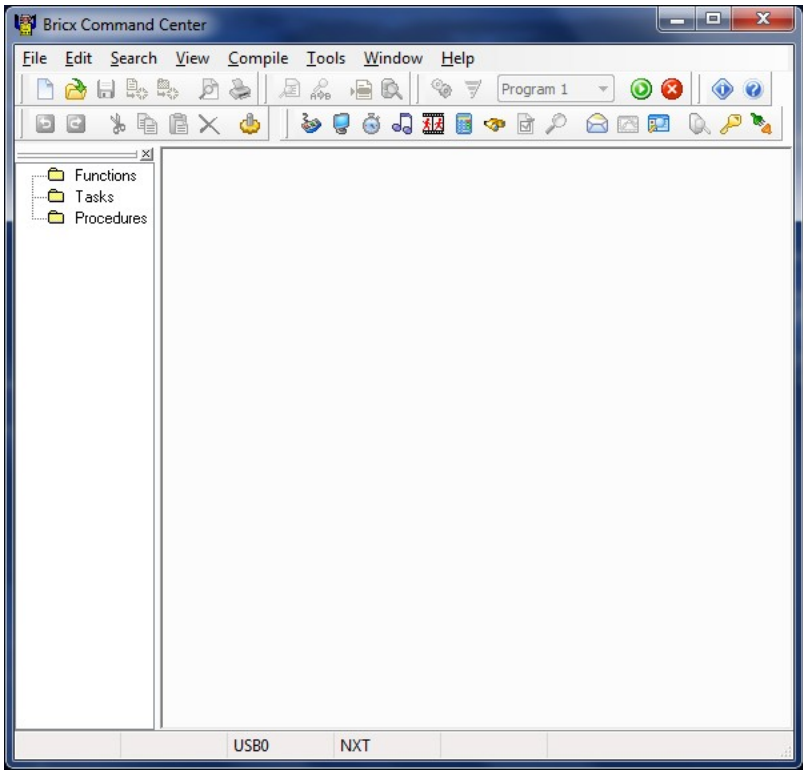

Рис.1.2. Интерфейс программы BricxCC

Интерфейс программы похож на интерфейс многих программ для Windows и имеет в том числе кнопки для создания и открытия файлов, открытия файлов с “помощью” по программе и многие другие. Однако есть несколько специальных кнопок для загрузки программы в NXT Brick, запуска программы на исполнение и др.

Для создания новой программы необходимо зайти в меню “Files” и нажать вкладку **New**, либо нажать Ctrl+N. *Без сохранения файла программа BricxCC будет выдавать сообщение про ошибку при компиляции.*

Вывод информации на дисплей

В качестве первой программы для освоения работы с дисплеем предлагается одна из самых популярных программ для проверки работы дисплея:

```
task main()
{
TextOut(20, LCD_LINE5, "Hello world");   //Вывод текста в пятой строке
Wait(15000);                             //Ожидание 15 секунд
}
```

Данный код необходимо набрать в рабочей области программы, и, нажав на синюю иконку (см. Рис.1.3. на панели программы) загрузить в NXT Brick. Запустить выполнение программы можно нажав зелёную кнопку (Рис.1.3.), либо посредством оранжевой квадратной кнопки непосредственно на NXT Brick.

Рис.1.3. Кнопки меню, необходимые для загрузки и запуска загруженной программы на выполнение

В результате выполенения программы в середине LCD дисплея NXT Brick, имеющего разрешение 100x64 пикселей, должны отобразиться слова: “**Hello world**”.

Данная программа состоит из “задачи” **main**, которая должна присутствовать в абсолютно каждой программе, функции **TextOut**, предназначенной для вывода текста, и функции **Wait**, предназначенной для введения задержки.

Функция **TextOut (int x, int y, string *str*)** в качестве аргументов имеет две целочисленных переменных, указывающих координаты **x** и **y** и

переменную-строку, содержащую выводимый на экран текст. В качестве координаты **y** может указываться номер строки как **LCD_LINE1**.

Функция **Wait(unsigned long int)** в качестве аргумента имеет только одну беззнаковую переменную, указывающую количество миллисекунд задержки. В рассматриваемом примере задержка будет составлять 15000/1000=15 секунд.

В конце каждой строки ставятся точки с запятой, что обозначает окончание одного выражения и начало другого.

Очистку экрана можно осуществить с помощью функции **ClearScreen()**, а очистку определённой линии можно осуществить с помощью функции **ClearLine(LCD_LINE5)**.

Использование таймера

NXT Brick содержит таймер, непрерывно работающий во время работы блока, который удобно использовать в некоторых ситуациях. Например когда отсчёт необходимо вести от одного и того же момента времени (для сравнения функция **Wait** ожидает заданное количество миллисекунд от момента использования функции).

Считать текущее значение таймер можно с помощью функции **CurrentTick()**. Приравняв эту функцию слева к какой-нибудь переменной можно сохранить текущее значение таймера (**time = CurrentTick()**).

Следующая программа содержит пример использования таймера:

```
task main()
{
long time;                          //Задание переменной типа long
time = CurrentTick();               //Сохранение текущего времени
TextOut(0, LCD_LINE5, "Timer is working");
until ((CurrentTick()-time) > 3000);    //Работать до истинности выражения,
                                        //т.е. пока не пройдёт 3 секунды
ClearLine(LCD_LINE5);               //Очистить линию 5
TextOut(10, LCD_LINE6, "Time is over");
until ((CurrentTick()-time) > 5000);
}
```

В результате выполнения программы на пятой строке дисплея в течении 3-х секунд будет отображаться текст **“Timer is working”**, после чего пятая строка будет очищена и на шестой строке 2 секунды будет отображаться текст **“Time is over”**.

Использование функции **until()** позволяет выполнять предыдущее действие до наступления истинности выражения в скобках.

Воспроизведение звуков

Встроенный динамик может использоваться для подачи сигнала о достижении некоторой цели, либо для создания музыкального сопровождения в ходе выполнения другого задания.

Воспроизводить звуки через встроенный динамик можно с помощью функции **PlayToneEx (unsigned int *frequency*, unsigned int *duration*, byte *volume*, bool *loop*)**.

Параметры функции **PlayToneEx** отвечают за следующие особенности воспроизведения:

frequency – частота звука;

duration – длительность в миллисекундах;

volume – громкость (0...4, где 4 – самая высокая громкость);

loop – логическая переменная типа bool, определяющая, будет ли повторяться звук после окончания воспроизведения.

Для упрощения изменения задаваемых переменных программы часто пользуются конструкцией **#define**, позволяющей упростить процесс настройки программы. Особенно это удобно при большом количестве однотипных переменных; например отвечающих за громкость.

Следующая программа содержит пример использования функции для вопроизведения звуков и применения конструкции **#define**:

```
#define VOL 3
task main()
{
PlayToneEx(262,400,VOL,FALSE); Wait(500);   //воспроизведение в течение 400 мс
                  //звука, имеющего частоту 262 Гц, с громкостью 3, без повторения
PlayToneEx(294,400,VOL,FALSE); Wait(500);
PlayToneEx(330,400,VOL,FALSE); Wait(500);
PlayToneEx(294,400,VOL,FALSE); Wait(500);
PlayToneEx(262,1600,VOL,FALSE); Wait(2000);
}
```

Благодаря использованию конструкции **#define** громкость всех функций можно изменить очень быстро, записав рядом с **VOL** другое значение громкости, например 1.

Для примера предлагается запустить мелодию “Имперского марша” из фильма “Звёздные войны”. Мелодия запускается при нажатии на правый треугольник, расположенный на блоке NXT Brick.

```
#define VOL 3
#define wait_time 600
sub Imperial_march()
{
long length=500;                    //длительность звука
for (int i=0;i<2;i++)        //воспроизводить два раза
{
PlayToneEx(392, length, VOL,FALSE); Wait(wait_time);
PlayToneEx(392, length,VOL,FALSE); Wait(wait_time);
PlayToneEx(392, length,VOL,FALSE); Wait(wait_time);
PlayToneEx(311, length *0.75,VOL,FALSE); Wait(wait_time*0.75);
PlayToneEx(466, length *0.25,VOL,FALSE); Wait(wait_time*0.25);
          PlayToneEx(392, length,VOL,FALSE); Wait(wait_time);
          PlayToneEx(311, length *0.75,VOL,FALSE); Wait(wait_time*0.75);
          PlayToneEx(466, length *0.25,VOL,FALSE); Wait(wait_time*0.25);
          PlayToneEx(392, length *2,VOL,FALSE); Wait(wait_time*2);
PlayToneEx(587, length,VOL,FALSE); Wait(wait_time);
PlayToneEx(587, length,VOL,FALSE); Wait(wait_time);
PlayToneEx(587, length,VOL,FALSE); Wait(wait_time);
PlayToneEx(622, length *0.75,VOL,FALSE); Wait(wait_time*0.75);
PlayToneEx(466, length *0.25,VOL,FALSE); Wait(wait_time*0.25);
          PlayToneEx(369, length,VOL,FALSE); Wait(wait_time);
          PlayToneEx(311, length *0.75,VOL,FALSE); Wait(wait_time);
          PlayToneEx(466, length *0.25,VOL,FALSE); Wait(wait_time*0.25);
          PlayToneEx(392, length *2,VOL,FALSE); Wait(wait_time*2);
}
}

task  main()
{
 long time;
 time=CurrentTick();
 while(CurrentTick()-time<100000)            //работать не более 100с
 {
  if (ButtonPressed(BTNRIGHT, true)==1)  //запуск мелодии нажатием правой кнопки
  {
   TextOut(25,LCD_LINE5,"Star wars");
   Imperial_march();                                 //проигрывание мелодии
   Wait(100);
   ClearScreen();
   }
   if (ButtonPressed(BTNRIGHT, true)==0)
   {
   TextOut(30,LCD_LINE5,"Waiting");
   Wait (100);
   ClearScreen();
   }
 }
}
```

Задания для самостоятельной работы:

1. Вывести в центре 2-й и 6-й строки латиницей имя и фамилию одного из авторов программы.

2. С использованием таймера отображать на дисплее 3 секунды имя одного из авторов программы, а потом в течении 3-х секунд фамилию автора в другой строке.

3. Выводить имя автора сначала в первой строке, потом во второй, ..., в последней строке (по кругу).

4. Выводить имя и фамилию одного из авторов по одной букве (а не полностью сразу).

5. При нажатии стрелок “влево” выводить на дисплее имя с фамилией одного автора, а при нажатии “вправо” имя второго автора (либо соседа).

6. Составить свою мелодию, изменяя частоту и длительность воспроизведения звуков в программе из примера.

7. При нажатии стрелок на блоке NXT Brick запускать воспроизведение либо мелодии из задания 4, либо мелодии из примера.

Лабораторная работа №2

….
….
….