БГУИР

# Кафедра ЭВМ

### Отчет по лабораторной работе № 1

**Тема: «Энергопитание»**

Выполнил:

студент группы 130501 Гнездилов А.М.

Проверила:

ассистент Игнатович А.О.

#### Минск 2023

1. **ЦЕЛЬ РАБОТЫ**

Целями лабораторной работы является вывести информацию об энергопитании компьютера:

- в режиме реального времени показывать тип энергопитания;

- информация о типе батареи;

- уровень заряда батареи (%);

- текущий режим энергосбережения;

- реализовать переход в спящий режим и гибернацию из консоли.

- вывести время работы аккумулятора без подключения к зарядке;

- вывести оставшееся время работы аккумулятора после отключения зарядного устройства.

1. **ЛИСТИНГ КОДА**

#include <Windows.h>

#include <iostream>

#include "locale.h"

#include <Powrprof.h> //Для работы SetSuspendState

using namespace std;

int main()

{

setlocale(LC\_ALL, "RUS");

SYSTEM\_POWER\_STATUS powerStatus; // переменная для хранения информации

int b = 0;

while (true) {

cout << "1 - Информация об энерпитании" << endl;

cout << "2 - Смена режима работы" << endl;

cout << "3 - Выход" << endl;

cin >> b;

switch (b) {

// Получаем информацию о состоянии батареи

case 1:

if (!GetSystemPowerStatus(&powerStatus)) //

{

cout << "Ошибка при получении информации о состоянии батареи\n";

return 1;

}

// Выводим информацию о состоянии батареи

cout << "Состояние батареи: ";

switch (powerStatus.ACLineStatus) // значение указывающее на подключение зарядки (Структура System Power Status)

{

case AC\_LINE\_OFFLINE:

cout << "отключена от зарядки\n";

break;

case AC\_LINE\_ONLINE:

cout << "подключена к зарядке\n";

break;

default:

cout << "неизвестно\n";

break;

}

//Уровень заряда батареи

cout << "Уровень заряда батареи: " << static\_cast<int>(powerStatus.BatteryLifePercent) << "%\n";

//Время заряда на 100%

if (powerStatus.BatteryFlag == 8) { //состояние батареи в битах(0 - батарея присутсвует, 1 - заряжается, 8 - заряжена)

cout << "Батарея заряжена на 100%." << endl;

}

else if (powerStatus.BatteryFlag == 128) {

cout << "Батарея отсутсвует." << endl;

}

else if (powerStatus.BatteryLifePercent > 0) {

int batteryLifeTime = (powerStatus.BatteryLifeTime / 60) \* 100 / powerStatus.BatteryLifePercent;

int hours = batteryLifeTime / 60;

int minutes = batteryLifeTime % 60;

cout << "При 100% заряда батареи: " << hours << " hours " << minutes << " minutes" << endl;

}

//Тип батареи(если 9, то бит 00х8(полностью заряжена) и 00х1(батарея присутствует))

cout << "Тип батареи: " << static\_cast<int>(powerStatus.BatteryFlag) << endl;

// Выводим оставшееся время зарядки

if (powerStatus.BatteryFlag != 128) // Проверка на наличие батареи

{

if (powerStatus.BatteryLifeTime != -1) //оставшееся время

{

int hours = powerStatus.BatteryLifeTime / 3600;

int minutes = (powerStatus.BatteryLifeTime % 3600) / 60;

cout << "Оставшееся время работы от батареи: " << hours << " часа(ов) " << minutes << " минут" << endl;

}

else

{

cout << "Система не использует батарею" << endl;

}

}

else

{

cout << "Батарея отсутствует" << endl;

}

//Проверка выбора энергосбережения

cout << "Текущий режим энергосбережения: ";

if (powerStatus.SystemStatusFlag & 0x01) //состояние системы(0х01 - резервное питание, 0х02 - критический уровень, 0х03 - не обнаружена, 0х08 - не используется)

{

cout << "Режим энергосбережения" << endl;

}

else

{

cout << "Режим работы" << endl;

cin.ignore();

cin.get();

} break;

case 2:

//Режим сна и гибернации

cout << "1.Решим сна" << endl;

cout << "2.Решим гибернации" << endl;

int choice;

cin >> choice;

switch (choice) {

case 1:

if (SetSuspendState(FALSE, FALSE, FALSE) == FALSE) {

cout << "Ошибка при переходе в режим сна" << endl;

}

break;

case 2: //нет доступа к гибернации

if (SetSystemPowerState(TRUE, FALSE) == FALSE) {

cout << "Ошибка при переходу в режим гибернации" << endl;

}

break;

}

break;

case 3:

return 0;

}

}

return 0;

}

1. **ВЫВОД**

Энергопитание является крайне важным элементом, обеспечивающим работу компьютера, и своевременное обслуживание блока питания позволяют увеличить срок его службы, уменьшить вероятность возникновения проблем с питанием ПК.