Міністерство освіти і науки України

Черкаський державний технологічний університет

Кафедра інформаційної безпеки та комп’ютерної інженерії

**З В І Т**

**з лабораторної роботи № 4**

**на тему: «Програмування зовнішніх пристроїв»**

**з дисципліни: «Архітектура комп’ютера»**

ПЕРЕВІРИВ: ВИКОНАВ:

ст. викладач студенти групи КМ-175

Гресько С.О. Косенко Андрій

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_

(оцінка) ( підпис)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

( дата, підпис)

ЧЕРКАСИ 2018

**Тема:** Програмування зовнішніх пристроїв.

**Мета роботи:** Ознайомитися зі способами організації взаємодії процесора і зовнішніх пристроїв (ЗП) у складі ЕОМ.

**Теоретичні відомості**

Зв'язок процесора і ЗП може здійснюватися в синхронному або асинхронному режимі. Синхронний режим використовується для ЗП, завжди готових до обміну. У нашій моделі такими ЗП є дисплей і тоногенератор - процесор може звертатися до цих ЗП, не аналізуючи їх стан (правда дисплей блокує отримання даних після введення 128 символів, формуючи прапор помилки).

Асинхронний обмін передбачає аналіз процесором стану ЗП, яке визначає готовність ЗП видати або прийняти дані або факт здійснення деякої події, контрольованої системою. До таких пристроїв в нашій моделі можна віднести клавіатуру і блок таймерів.

Аналіз стану ВУ може здійснюватися процесором двома способами:

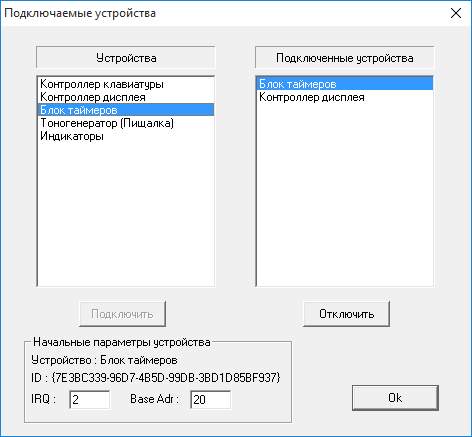
* в програмно-керованому режимі;
* в режимі переривання.

У першому випадку передбачається програмне звернення процесора до регістру стану ЗП з подальшим аналізом значення відповідного розряду слова стану. Таке звернення слід передбачити в програмі з деякою періодичністю, незалежно від фактичного настання контрольованої події (наприклад, натискання клавіші).

У другому випадку при виникненні контрольованого події ЗП формує процесору запит на переривання програми, за яким процесор і здійснює зв'язок з ЗП.

**ВИКОНАННЯ РОБОТИ**

1) Запустили програмну модель ЕОМ і підключили до неї ВУ Дисплей і Блок таймерів.



2) Згідно з завданням, нам потрібно вивести три повідомлення на дисплей, тому спочатку визначилися з текстом і кодами повідомлень:

|  |  |  |
| --- | --- | --- |
| Текст на дисплеї | Коди ASCII (табл. 8.7) | Коди в дес. виді (табл. 8.8) |
| Fi | 46 69 | 70 105 |
| Se | 53 65 | 83 101 |
| Th | 54 68 | 84 104 |

3) Написали і налагодили програму, передбачену завданням, в режимі програмного контролю. Отримаємо наступний код:

|  |  |
| --- | --- |
| Мнемокод | Коментар |
| **Перше повідомлення** | |
| CALL M | Сброс дисплею |
| RD #70  OUT 10 | Вивід на экран символа F  Вивід на экран символа i |
| RD #105  OUT 10 |
| **Перший таймер** | |
| RD #0  OUT 21 | Обнуляємо таймер |
| RDI 5001  OUT 22 | Установка ділителя в 5 і запуск таймера |
| M1: IN 21  SUB #200  JS M1  RDI 1000  OUT 22 | Цикл, в якому порівнюється поточне значення на таймері з порогом 200 (при дільнику 5 приблизно відповідає 15с), при досягненні якого зупиняємо таймер і рухаємося далі. |
| **Друге повідомлення** | |
| CALL M | Сброс першого повідомлення з дисплея |
| RD #83  OUT 10 | Вивід на экран символа S  Вивід на экран символа e |
| RD #101  OUT 10 |
| **Другий таймер** | |
| RD #0  OUT 21 | Обнуляємо таймер |
| RDI 5001  OUT 22 | Установка ділителя в 5 и запуск таймера |
| M2: IN 21  SUB #250  JS M2  RDI 1000  OUT 22 | Цикл, в якому порівнюється поточне значення на таймері з порогом 250 (при дільнику 5 приблизно відповідає 20с), при досягненні якого зупиняємо таймер і рухаємося далі. |
| **Третє повідомлення** | |
| CALL M | Сброс другого повідомлення з дисплея |
| RD #84  OUT 10 | Вивід на экран символа T  Вивід на экран символа h |
| RD #104  OUT 10 |
| HLT | Остановка програми |
| **Процедура сброса дисплею** | |
| M: RD #101  OUT 11 | Очищаємо дисплей і відеопам'ять зі скидами регістрів AR, Err і A |
| RD #011  OUT 11 | Встановлюємо регістри E і I в 1 для подальшої роботи з дисплеєм |
| RET | Повертаємося до основного коду |

**Переписали цей код з урахуванням обробки переривань, зациклені виконання програми:**

|  |  |
| --- | --- |
| Мнемокод | Коментар |
| **Перше повідомлення** | |
| EI  CALL M  RD #70  OUT 10  RD #105  OUT 10 | Початок обробки переривання  Очищення дисплея з виведенням на нього першого повідомлення (Fi) |
| **Друге повідомлення** | |
| RDI 99799  OUT 21  RDI 5001  OUT 22  M1: IN 21  JNZ M1 | Запуск першого таймера і очікування переповнення (без виклику переривання) |
| CALL M  RD #83  OUT 10  RD #101  OUT 10 | Очищення дисплея з виведенням другого повідомлення (Se) |
| **Третє повідомлення** | |
| RDI 99749  OUT 21  RDI 5001  OUT 22  M2: IN 21  JNZ M2 | Запуск другого таймера і очікування його переповнення (без виклику викликом переривання) |
| CALL M  RD #84  OUT 10  RD #104  OUT 10 | Очищення дисплея з виведенням третього повідомлення (Th) |
| RDI 99979  OUT 21  RDI 5011  OUT 22  M3: IN 21  JNZ M3  IRET | Короткий таймер з викликом переривання при переповненні перед поверненням до першого повідомлення |
| M: RD #101  OUT 11  RD #102  OUT 11  RD #011  OUT 11  RET | Процедура сброса дисплея |

***Висновок***: в ході виконання роботи було вивчено питання взаємодії процесора з зовнішніми пристроями (зокрема, з дисплеєм і таймерами) в складі навчальної моделі ЕОМ. Був складений код для вирішення завдання як в повністю програмному режимі управління, так і використанням переривань.