Міністерство освіти і науки України

Національний технічний університет України

«Київський політехнічний інститут»

Факультет інформатики та обчислювальної техніки

Кафедра обчислювальної техніки

ЛАБОРАТОРНА РОБОТА № 2

З дисципліни «Архітектура комп’ютерів-1»

На тему «Синтез блоків мікропрограмного управління»

ВИКОНАВ:

студент 2 курсу ФІОТ

групи ІО-41

Логвинчук А. І.

Залікова - 4120

Варіант – 0011000

ПЕРЕВІРИВ:

доц. Ткаченко В. В

Київ – 2016

**Завдання:**

Розробити БМУ для управління операційним пристроєм.

Функція: *F = X2* (2-ий спосіб множення).

Відносна адресація МК.

ПМК на 16 слів.

Контроль слова на непарність.

Тривалість МО підсумовування – 6 тактів.

**Розробка операційної схеми**

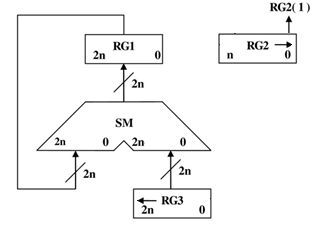


рис 1.1. Операційна схема пристрою множення другим способом

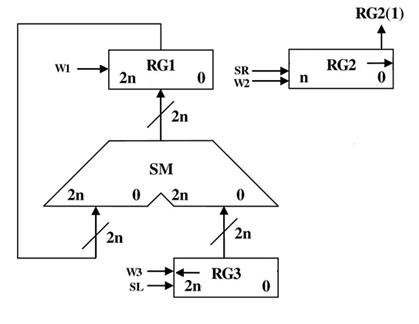
**Розробка функціональної схеми із зазначенням управляючих сигналів**

рис 1.2. Функціональна схема пристрою множення другим способом

W1 – запис n-частинної суми до регістру результату.

W2 – запис множника до регістру RG.2.

W3 – запис множеного до регістру RG.3.

SL – зсув машинного слова у регістрі RG.3 вліво.

SR – зсув машинного слова у регістрі RG.2 вправо.

**Змістовний структурний мікроалгоритм:**



рис. 1.3 Змістовний структурний мікроалгоритм

*Таблиця 1.1.* Логічне моделювання

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| №  такту | RG.2 |  | RG.3 | RG.1 | Мікрооперація |
| П. стан | 000 | 0 | 0000000 | 00000000 |  |
| 1 | 010 | 1 | 0000000 | 00000011 | RG.1:=0, RG.2:=Y, RG.3:=X |
| 2 | 010  001 | 1  0 | 00000000  +00000011 | 00000011  00000110 | RG.1:=RG.1+RG.3, RG.3:=l[RG.3].0, RG.2:=0.r[RG.2] |
| 3 | 001  000 | 0  1 | 000000011 | 00000110  00001100 | RG.3:=l[RG.3].0, RG.2:=0.r[RG.2] |
| 4 | 000  000 | 1  0 | 00000011  +00001100 | 00001100  00011000 | RG.1:=RG.1+RG.3, RG.3:=l[RG.3].0, RG.2:=0.r[RG.2] |
| Рез. |  |  | 00001111 |  |  |

*Таблиця 1.2.* Кодування

управляючих сигналів

|  |  |
| --- | --- |
| Управляючий сигнал | Код |
| W2 | Y1 |
| W3 | Y2 |
| W1 | Y3 |
| SR | Y4 |
| SL | Y5 |

*Таблиця 1.3* Кодування

логічних умов

|  |  |
| --- | --- |
| Логіна умова | Код |
| RG.2 = 0 | X1 |
| RG.2(1) | X2 |

**Закодований структурний мікроалгоритм**



рис. 1.4 Закодований структурний мікроалгоритм

**Синтез блока мікропрограмного управління.**

БМУ є блоком управління з гнучкою логікою: зміна мікроалгоритму не потребує модифікації БУ.

Для того щоб синтезувати БМУ необхідно розробити зонну структуру мікрокоманди.

**Структура зони β1**

Зона β1 у БМУ з відносною адресацією містить зміщення адреси наступної команди відносно даної та управляючі сигнали для мультиплексора.

Нехай максимальне зміщення дорівнює 15. Тоді для зміщення адреси потрібно log215 + 1 = 5 розрядів.

Для 2 логічних умов достатньо 2 розрядів управління мультиплексором.

Таким чином, довжина зони β1 становить 5+2=7 розрядів.

**Структура зони β2**

Для кодування управляючих сигналів доцільно використати комбінований спосіб: поєднати сигнали у групи таким чином, щоб сигнали, які мають бути виконані водночас розміщувались в різних групах:

*Таблиця 1.4. Комбіноване*

*групування сигналів*

|  |  |
| --- | --- |
| I | II |
| Y1 | Y2 |
| Y3 | Y5 |
| Y4 |  |

Для сигналів Y1, Y3, Y4 використаємо дешифратор. Завдяки цьому ми мінімізуємо кількість розрядів у зоні β2 та унеможливимо одночасне виконання несумісних у часі мікрооперацій. Аналогічно для другої групи.

Отже для зони β2 потрібно 4 розряди.

**Структура зони β3**

Зона β3 визначає затримку управляючих сигналів. В даному випадку максимальна затримка Δt = 6 – 1 = 5 тактів для операції підсумовування (Y3).

Зона β3 реалізується у вигляді лічильника. При обнуленні лічильника відбувається перехід на наступну команду. Число тактів затримки записується в лічильник у вигляді від’ємного в числа доповнювальному коді. Число 5 можна закодувати 3 розрядами, отже для зони β3 потрібно взяти 4 розряди для врахування знаку.

Отже розрядність зони β3 дорівнює 4.

**Структура зони β4**

Зона β4 використовується для контролю обладнання. В даному випадку ми перевірятимемо коректність запису мікрокоманди в РМК: кількість одиниць у записі слова має бути непарною.

**Програмування БМУ**

Розмістимо МК у пам’яті:

Таблиця 1.5. Розміщення МК у пам’яті

|  |  |
| --- | --- |
| Адр. МК | № МК |
| 0000 | П |
| 0001 | 1 |
| 0010 | 2 |
| 0011 | - |
| 0100 | 3 |
| 0101 | К |
| 0110 | 4 |
| 0111 | 5 |

Альтернативні команди повинні відрізнятись лише молодшим розрядом.

Запрограмуємо ПМК для виконання даного мікроалгоритму.

*Таблиця 1.6. Карта програмування БМУ*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| № | ADDR | β1 | | β2 | β3 | β4 |
|  |  | ЗР S4S3S2S1 | M2M1 | A4A3A2A1 | ЗР D3D2D1 | P |
| П | 0000 | 0.0001 | 00 | 0000 | 0.000 | 0 |
| 1 | 0001 | 0.0001 | 00 | 0110 | 0.000 | 0 |
| 2 | 0010 | 0.0010 | 01 | 0000 | 0.000 | 1 |
| 3 | 0100 | 0.0010 | 10 | 0000 | 0.000 | 1 |
| 4 | 0110 | 0.0001 | 00 | 1000 | 1.100 | 1 |
| 5 | 0111 | 1.1011 | 00 | 1101 | 0.000 | 0 |
| К | 0101 | 0.0000 | 00 | 0000 | 0.000 | 1 |

**Висновок**:

БМУ використовують для управління АЛП. БМУ мають гнучку логіку, таким чином дозволяючи змінювати мікроалгоритм без модифікації самого управляючого пристрою. Під час виконання мікропрограми в кожному такті із постійної пам'яті БМУ зчитується та розшифровується чергова мікрокоманда. В результаті виконання мікрокоманди формуються управляючі сигнали необхідної тривалості, що поступають на всі функціональні частини обчислювальної системи, а також формується адреса наступної мікрокоманди.