Національний технічний університет України

«Київський політехнічний інститут»

Факультет інформатики та обчислювальної техніки

Кафедра обчислювальної техніки

Лабораторна робота № 6. Проектування CPU

з дисципліни «Комп’ютерна схемотехніка»

Виконав студент групи ІО-01 *Редько Олександр*

Номер залікової книжки **10310**

# Завдання

На PLMT с параметром  и раннее разработанных ICTR, LSM, RAM, FM и AU разработать вычислитель, реализующий команды +, -, пересылки (RAM - FM), условных и безусловных переходов, команды ввода-вывода.

# Визначення варіанту

 ‑ 7 входов.

# Виконання роботи

## Построение структурной схемы CPU



Рисунок 3.1 – Структурная схема CPU

## Параметры CPU

**RAM** – 8K x 15

Адрес обращения к RAM 13-битный

Записью управляет сигнал WR

**FM** – 3 однонаправленные шины (AQ,AB чтение, AD запись), 15х15

Записью управляет сигнал WR

**LSM** управляется кодом микрооперации F0F1F2 (3 бита, т.к. 8 МО)

**SHU>** реализует сдвиг вправо для функциональной полноты AU

Сигнал F> управляет сдвигом

**ICTR** счетчик команд, т.к. длина команды 15 бит, счётчик увеличивается на +2

**IRG** регистр команд, состоит из 4 байт

**COP** блок управления операциями (формирует управляющие сигналы Сi)

**CCRG** регистр признаков операции (сохраняет признаки SI,RZ,CO,Q0)

## Кодирование и структура команд CPU

**AO/LO** 4 бита кода МО LSM (3 бита F0F1F2 для LSM и один F> для SHU>)

3 5-битных адреса операндов FM

23 x 21 x 25 x 25 x 25 = 219 разновидностей операций

**FR/RF** 1 бит для направления пересылки

8-битный адрес ячейки в FM

13-битный адрес ячейки в RAM

21 x 28 x 213 = 222 разновидностей операций

**JC** 2 бита для условий (RZ, SI, CO, QO) и 1 бит для их невыполнения

12 бит адреса перехода в RAM

23 x 213 = 216 разновидностей операций

**JMP** 13 бит адреса перехода в RAM

213 разновидностей операций

**IN/OUT** 1 бит для направления передачи

13 бит для номера устройства (предположим, что количество устройств не больше ёмкости RAM)

21 x 213 = 214 разновидностей операций

219 + 218 + 215 + 213 + 213 > 222

Отсюда следует, что для задания команды можно использовать 3 байта. Однако 3 байта дают возможность кодировать 224 команд. Следовательно, такое кодирование может быть избыточным.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| OP | | BY0 | | | | | | | | BY1 | | | | | | | | BY2 | | | | | | | | BY3 | | | | | | | |
|  | | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| JMP | | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | A RAM | | | | | | | | | | | |  |  |  |  |  |  |  |  |
| JC | | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | A CC | | | A RAM | | | | | | | | | | | |  |  |  |  |  |  |  |  |
| FR/RF | | 1 | 1 | 1 | 1 | 1 | 0 | 0/1 | A FM | | | | | A RAM | | | | | | | | | | | |  |  |  |  |  |  |  |  |
| IN/OUT | | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0/1 | A RAM | | | | | | | | | | | |  |  |  |  |  |  |  |  |
| AO | 0 | 1 | 0 | 0 | 0 | 0 | AQ | | | | | AB | | | | | AD | | | | | F0F1F2 | | |  |  |  |  |  |  |  |  |  |
|  | 1 | 1 | 0 | 0 | 1 | 0 |  |  |  |  |  |  |  |  |  |
|  | CO | 1 | 0 | 1 | 0 | 0 |  |  |  |  |  |  |  |  |  |
|  | SI | 1 | 0 | 1 | 1 | 0 |  |  |  |  |  |  |  |  |  |
|  | Q0 | 1 | 1 | 0 | 0 | 0 |  |  |  |  |  |  |  |  |  |
| LO |  | 1 | 0 | 0 | 0 | 1 |  |  |  |  |  |  |  |  |  |
| SH> | 0 | 0 | 0 | 0 | 0 | 1 | AQ | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | 1 | 0 | 0 | 0 | 1 | 1 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | CO | 0 | 0 | 1 | 0 | 1 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | SI | 0 | 0 | 1 | 1 | 1 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | Q0 | 0 | 1 | 0 | 0 | 1 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

АСС ­­­­– код для условия перехода

ARAM – адрес ячейки в RAM

AFM– адрес ячейки в RAM

АО – признак BY00 = 1, BY04 = 0

LО – признак BY00 = 1, BY02 = 0, BY04 = 1

SH> – признак BY00 = 0, BY04 = 1

## Кодирование условий СС

0 000

1 001

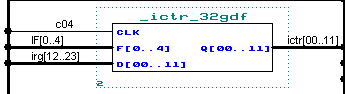
CO 010

SI 011

Q0 100

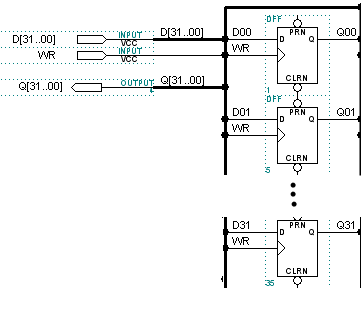
## ICTR

Счетчик на 12 разрядов команд, инкрементирует по +2.



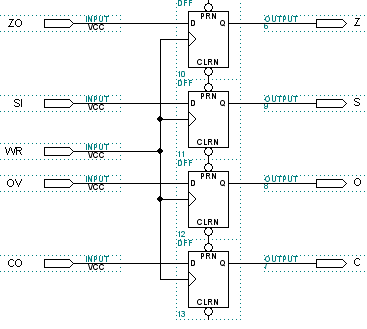
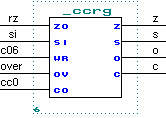
## IRG

Регистр команд на 15 бит.



## CCRG

Слово состояния должно учитывать 4 флага состояния, формируемых после каждой операции в арифметическом устройстве SI, RZ, CO, Q0, где SI – признак знака, RZ – признак ноля в результате, CO – признак переполнения на сумматоре, Q0 – разряд, теряемый при сдвиге на сдвигателе. Слова состояния будут перезаписываться при каждом выполнении команды из первой группы команд.

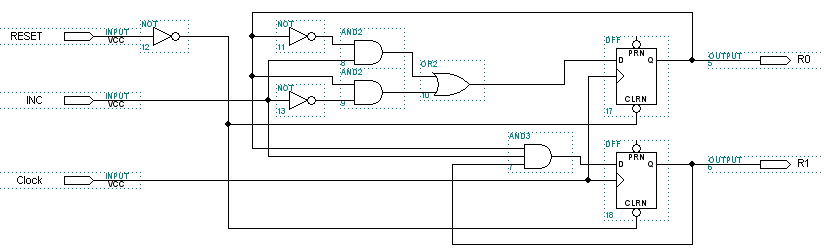
## CTR

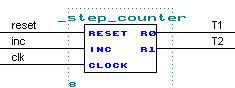
**Диаграмма тактов**

|  |  |  |  |
| --- | --- | --- | --- |
|  | T1 | T2 |  |
| JMP | C3,C5 | C4 | M1 |
| JC | C3,C5 | C4 CC / C3 | M2 |
| RF | C3,C5 | C1, C7 | M3 |
| FR | C3,C5 | C1,C2,C10 | M4 |
| IN | C3,C5 | C7,C9 | M5 |
| OUT | C3,C5 | OUT | M6 |
| AO/LO | C3,C5 | C8,C10,C6 | M7,M8 |
| SH> | C3,C5 | C8,C10,C6 | M9 |

**Кодирование тактов**

|  |  |
| --- | --- |
| T1 | T2 |
| 1 | 0 |
| 0 | 1 |





## COP

Из таблицы тактов следует, что управляющие сигналы Сi должны формироваться по выражениям:

C1 = T2M4 v T2M3

C2 = T2M4

C3 = T1 v T2M2

C4 = T2M1 v T2M2 Cc

C5 = T1

C6 = T2M7 v T2M8 v T2M9

C7 = T2M5

C8 = T2M7 v T2M8 v T2M9

C9 = T2M5

C10 = T2M4 v T2M7 v T2M8 v T2M9

При этом Мi – дешифрированный код ОР, а именно:











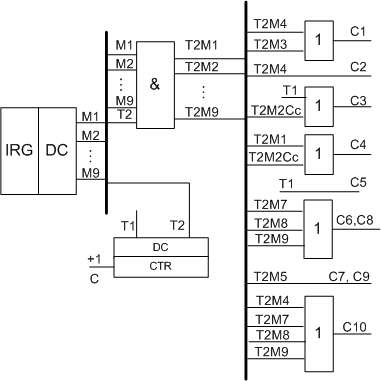








В таком случае, COP должен содержать набор ЛЭ И-ИЛИ, DC кодов ОР и CTR тактов. На 10 И формируются произведения T2Mi (i = 1,10). Далее выходы И собираются на ИЛИ в соответствии с формулами для управляющих сигналов. Предполагается, что произведения T2M2 и T2M2 Cc формируются в 2 этапа: 1) в схеме получают сигнал T2M2. 2) после опроса Сс формируют сигналы T2M2 и T2M2 Cc

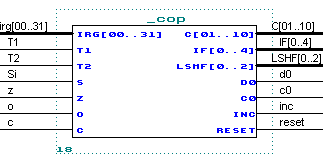


Но кроме того, COP формирует такие сигналы:

LSHF[0..2] – управляющие сигналы для LSM (F0F1F2)

Inc, reset – инкремент и сброса счетчика тактов

d0, c0 – вдвигаемые разряды для LSM и SHU>

****