Описание: Kpi-best

Міністерство освіти і науки, молоді та спорту України

Національний технічний університет України “Київський політехнічний інститут”

Факультет інформатики та обчислювальної техніки

Кафедра обчислювальної техніки

Лабораторна робота № 2

з курсу “Комп’ютерної схемотехніки” на тему:

“Проектування LSM”

Виконав: студент ІІІ курсу групи ІО – 01

Рудницький М.В.

Перевірили: доцент кафедри ОТ Корнійчук В.І.

старший викладач кафедри ОТ Долголенко О.М.

Київ – 2012

**Завдання**: на PLMT с параметром построить n-разрядный LSM () с частично групповым переносом и заданным набором из операций: , , , , , , , , , , 0. LSM должен содержать узлы, формирующие признаки RZ (нулевой результат), SI (знак), CO (перенос из старшего разряда). Оценить сложность полученной схемы и ее быстродействие.

**Визначення варіанту:**

– входа

– разрядность LSM

– операций

**Виконання роботи:**

Рассмотрим LSM как совокупность одноразрядных LSM. Тогда его составные:

1. Одноразрядный сумматор по модулю 2 (2К-НЕ).
2. Формирователь переноса (CI).
3. Логические схемы управления подаваемыми операндами LNQ.
4. Логические схемы управления подаваемыми операндами LNP.

Структурная схема представлена на рисунке 1.

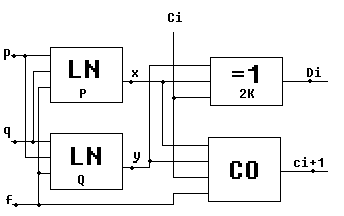


Рисунок 1 – Структурная схема LSM.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  | |  |  |  |
|  | 0 | 1 | 0 | 1 |  |
|  | 1 | 0 | 1 | 0 |  |
|  |  |  | |  |  |

Рисунок 2 – Диаграмма Вейча.

**Проектирование сумматора по модулю 2 на 3 входа**

Построим диаграмму Вейча, представленную на рисунке 2. Очевидно, что данная функция не минимизируется.

Реализация функции:  представлена на рисунке 3.

Рисунок 3 – Реализация функции.

Таблица 1 – Таблица истинности.

|  |  |  |  |
| --- | --- | --- | --- |
| X | Y | C | **D** |
| 0 | 0 | 0 | **0** |
| 0 | 0 | 1 | **1** |
| 0 | 1 | 0 | **1** |
| 0 | 1 | 1 | **0** |
| 1 | 0 | 0 | **1** |
| 1 | 0 | 1 | **0** |
| 1 | 1 | 0 | **0** |
| 1 | 1 | 1 | **1** |

## Кодирование элементов и составление таблиц истинности

Выберем коды и передаваемые значения так, чтобы кодирующие функции имели простой вид (таблица 2). При этом 

Таблица 2 – Кодирование функций.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **F3** | **F2** | **F1** | **F0** | **Функция** | **Xi** | **Yi** | **Ci + 1** | **C0** |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 |  |  | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 |  |  | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 |  |  | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 |  | P | Q | T | 0 |
| 0 | 1 | 0 | 1 |  | P |  | T | 1 |
| 0 | 1 | 1 | 0 |  | P | P | T | 0 |
| 0 | 1 | 1 | 1 |  | P | 0 | T | 1 |
| 1 | 0 | 0 | 0 |  | P | Q | 0 | 0 |
| 1 | 0 | 0 | 1 |  |  | 0 | 0 | 0 |
| 1 | 0 | 1 | 0 |  | P |  | 0 | 0 |
| 1 | 0 | 1 | 1 | x | x | x | x | 0 |
| … | … | … | … | … | … | … | … | ... |

Приведем диаграммы Вейча для наглядности минимизации функций (рисунок 4).

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  | |  |  |  | *Xi* |
|  |  |  | |  |  |  |  |
|  |  | x | x | C:\Documents and Settings\Мирослав\Рабочий стол\notp.bmp | P |  |  |
|  | x | x | x | P |  |  |
|  |  | P | P | C:\Documents and Settings\Мирослав\Рабочий стол\porq.bmp | C:\Documents and Settings\Мирослав\Рабочий стол\pnotq.bmp |  |
|  |  | P | P | C:\Documents and Settings\Мирослав\Рабочий стол\notporq.bmp | 0 |  |  |
|  |  |  |  | |  |  |  |
|  |  |  |  | |  |  |  |
|  |  |  | |  |  |  | *Yi* |
|  |  |  | |  |  |  |  |
|  |  | x | x | 0 | Q |  |  |
|  | x | x | x | C:\Documents and Settings\Мирослав\Рабочий стол\pnotq.bmp |  |  |
|  |  | P | 0 | 0 | 0 |  |
|  |  | Q | C:\Documents and Settings\Мирослав\Рабочий стол\notq.bmp | 0 | 0 |  |  |
|  |  |  |  | |  |  |  |
|  |  |  |  | |  |  |  |
|  |  |  | |  |  |  | *Ci + 1* |
|  |  |  | |  |  |  |  |
|  |  | x | x | 0 | 0 |  |  |
|  | x | x | x | 0 |  |  |
|  |  | T | T | 0 | 0 |  |
|  |  | T | T | 0 | 0 |  |  |
|  |  |  |  | |  |  |  |
|  |  |  |  | |  |  |  |
|  |  |  | |  |  |  | *C0* |
|  |  |  | |  |  |  |  |
|  |  | 0 | 0 | 0 | 0 |  |  |
|  | 0 | 0 | 0 | 0 |  |  |
|  |  | 0 | 1 | 0 | 0 |  |
|  |  | 0 | 1 | 0 | 0 |  |  |
|  |  |  |  | |  |  |  |
|  |  |  |  | |  |  |  |

Рисунок 4 – Минимизация методом Вейча.

**Проектирование цепей переноса**

Xi – функция, передаваемая на сумматор и цепи переноса через LNP.

.

Yi – функция, передаваемая на сумматор и цепи переноса через LNQ.

*.*

Функции и *.*



Рисунок 5 – Реализация C0.

Рисунок 6 – Реализация Ci + 1.



Рисунок 7 – Реализация функции Xi.



Рисунок 8 – Реализация функции Yi.

**Формирователи признаков**

Для ухода от проблем потери знака вследствие переноса будем использовать модифицированный код. Он предполагает два знаковых разряда, с которыми мы имеем право совершать все арифметические действия. Следовательно, для формирования признака SI мы можем брать старший разряд результата при разрешенных переносах. То есть: . Признак переноса снимаем сразу с элемента цепи переносов CG11. Само построение цепи переносов не выдаст перенос при выполнении логических операций. Признак нулевого результата формируем исходя из того, что все разряды мантиссы (D0 – D7) должны быть нулевыми. Откуда: .



Рисунок 9 – Схема формирования признака SI.



Рисунок 10 – Схема формирования признака RZ.

**Общая схемная реализация**



Рисунок 11 – Реализация i-го разряда LSM.

**Оценка сложности и быстродействия**