

АНАЛИЗ И СИНТЕЗ НА ЛОГИЧЕСКИ СХЕМИ

Лекция

7.1. КОДОВИ ПРЕОБРАЗОВАТЕЛИ

7.1.1. Същност и определение

Схемите, предназначени за преобразуване на цифрова информация от десетична бройна система в двоична, обикновено се наричат шифриращи, а за обратно преобразуване – дешифриращи. Схемите, предназначени за преобразуване на различни кодове в двоична бройна система, се наричат преобразуватели на код или *кодови преобразуватели*. Дешифраторите и шифраторите представляват частен случай на преобразувателите на кодове.

Под преобразуване на кодове се разбира преобразуване на n - разрядни двоични числа, представляващи информация в един зададен код, в m - разрядни двоични числа, представляващи тази информация в друг код. Самият кодов преобразувател е комбинационна логическа схема, която преобразува даден входен двоичен код в друг, точно определен изходен код. При това броят на двоичните разряди на входния и изходния код може да не е еднакъв.

Двоично-десетичните-десетични преобразуватели са най-широко разпространени, тъй като цифровите устройства работят в двоичен или двоично-десетичен код, а цифровите резултати се извеждат на индикатори в десетичен код.

7.1.2. Синтез на кодов преобразувател

Съществуват три основни подхода за построяване на кодови преобразуватели:

- реализация по схема „дешифратор – шифратор“;
- непосредствена реализация на система от Булеви функции на група аргументи (чрез таблица);
- реализация с програмируеми логически матрици (ПЛИМ, PLA) и постоянни запомнящи устройства (ПЗУ).

В лекция 6 е показан първия от подходите за построение на кодов преобразувател, реализация по схема „дешифратор – шифратор“. Структурната схема на такъв кодов преобразувател е представена на фиг. 6.3.

При втория основен подход се използва таблица с всички възможни комбинации на входните променливи и с желаните резултати за всяка комбинация на входните променливи. Така кодовите преобразуватели могат да се синтезират върху основата на таблица, която дава съответствието между входния и изходния код. При това всеки разряд на изходния код се разглежда като логическа функция на променливите, чрез които се представя входния код. Ако във входния код има забранени (неработни, неизползвани,

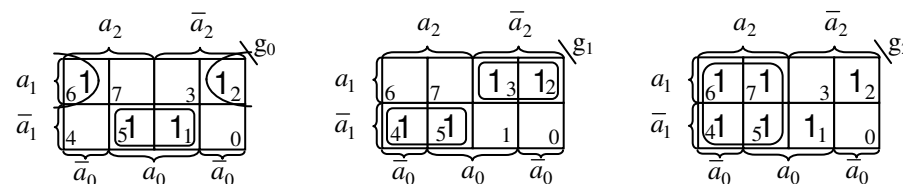
фиктивни) комбинации, те представляват неопределени набори за всички изходни функции и се използват за допълнително опростяване на изразите за изходния код.

Пример 7.1-1. Да се синтезира кодов преобразувател за преобразуване на двоичен еквивалент (двоичен код 421) в цикличен код на Грей.

За синтеза на кодовия преобразувател ще бъде използван вторият подход- (чрез таблица-таблица 7.1).

Таблица 7.1

Десетично число	Двоичен код 421			Код на Грей		
	a_2	a_1	a_0	g_2	g_1	g_0
0	0	0	0	0	0	0
1	0	0	1	0	0	1
2	0	1	0	0	1	1
3	0	1	1	0	1	0
4	1	0	0	1	1	0
5	1	0	1	1	1	1
6	1	1	0	1	0	1
7	1	1	1	1	0	0

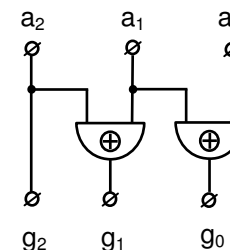


$$g_0 = \bar{a}_1 a_0 + a_1 \bar{a}_0 = a_1 \oplus a_0; \quad (7.1)$$

$$g_1 = \bar{a}_2 a_1 + a_2 \bar{a}_1 = a_2 \oplus a_1; \quad (7.2)$$

$$g_2 = a_2. \quad (7.3)$$

От намерените минимални форми на функциите g_0 (7.1), g_1 (7.2) и g_2 (7.3) се синтезира схемата на кодовия преобразувател (фиг. 7.1).



Фиг. 7.1. Кодов преобразувател за преобразуване на двоичен код 421 в код на Грей

7.2. КОМБИНАЦИОННИ СУМАТОРИ

7.2.1. Основни правила

Събирането в двоична бройна система се явява най-важната аритметична операция, тъй като то лежи в основата и на другите аритметични операции: изваждане, умножение, деление. Аритметичната операция събиране, извършваща се с двоични числа е подобна на действието събиране с десетични числа. При сумиране на две двоични цифри 1 се получава 0 и възниква така нареченият пренос 1, което означава предаване на цифра 1 в близкия (съседен) по висш (по старши) разряд, където тя се сумира с намиращите се там цифри. Основни правила на аритметичната операция събиране за случая с две двоични цифри (две едноразрядни числа) са следните:

$$0+0=0 \text{ (сума)} \text{ и } 0 \text{ (пренос в следващия по-старши разряд);} \quad (7.4)$$

$$0+1=1 \text{ (сума)} \text{ и } 0 \text{ (пренос в следващия по-старши разряд);} \quad (7.5)$$

$$1+0=1 \text{ (сума)} \text{ и } 0 \text{ (пренос в следващия по-старши разряд);} \quad (7.6)$$

$$1+1=0 \text{ (сума)} \text{ и } 1 \text{ (пренос в следващия по-старши разряд).} \quad (7.7)$$

При събиране на три двоични цифри 1 се получава 1 и 1 пренос:

$$1+1+1=1 \text{ (сума)} \text{ и } 1 \text{ (пренос в следващия по-старши разряд)} \quad (7.8)$$

7.2.2. Класификация на суматорите

Суматорът е функционален възел, изпълняващ операцията аритметично събиране на две числа. Основа за построяването на всеки суматор е едноразрядния суматор.

В зависимост от организацията на процеса на сумирането, суматорите се делят на:

- комбинационни суматори (КС);
- натрупващи суматори (НС);
- суматори от комбинационно-натрупващ тип (КНС).

В зависимост от броя на разрядите, участващи в даден момент в събирането, суматорите се делят на:

- едноразрядни;
- многоразрядни.

По начина на обработка на многоразрядните числа се различават:

- последователни;
- паралелни;
- последователно-паралелни.

В зависимост от организацията на веригата за преноса, суматорите се делят на:

- суматори с последователен пренос;
- суматори с паралелен пренос;
- суматори с групов пренос.

7.2.3. Комбинационни суматори

Основните схеми, обезпечаващи елементарните операции събиране са

така наречените едноразрядни суматори, които могат да бъдат пълни и непълни (полусуматори). Непълният суматор има само два входа и два изхода. Единият изход дава резултата от сумирането на двата бита, а вторият изход дава резултата за преноса, възникващ при сумирането. Полусуматорът се използва за събиране на две най-малко значещи цифри при двоично сумиране.

Пълният едноразряден суматор има три входа и два изхода. На третият вход се подава резултата за пренос от предходния (по-младшия) суматор. По-голямо значение имат пълните едноразрядни суматори, които могат да се съставят от два непълни.

7.2.3.1. Едноразрядни непълни суматори

Едноразрядните непълни суматори представляват комбинационни схеми за реализиране на операцията събиране на двоични цифри. Ако схемата има два входа и два изхода и реализира правилата за сумиране на две едноразрядни двоични числа, $0+0=0$, $0+1=1$, $1+0=1$, $1+1=0$ и пренос 1 в старшия разряд, се нарича едноразряден непълнен суматор (полусуматор).

Таблица 7.3-1.

Таблица на истинност			
x_i	y_i	P_i	S_i
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

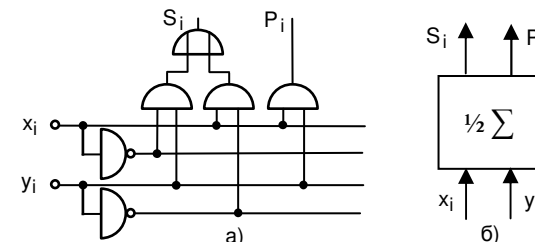
Таблица 7.3-1 е таблицата на истинност на полусуматора, от която се получават изходните логически функции сума S_i и пренос P_i .

$$S_i = \bar{x}_i y_i + x_i \bar{y}_i = x_i \oplus y_i; \quad (7.9)$$

$$P_i = x_i y_i. \quad (7.10)$$

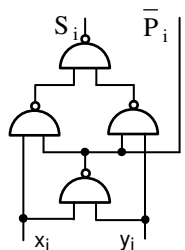
Функциите S_i (7.9) и P_i (7.10), са представени в канонична форма. Те са познати елементарни логически функции и не могат да се опростят допълнително. Означенията S_i и P_i са съответно *сума* и *пренос*, а x_i и y_i са едноименните разряди на двете двоични числа X и Y , които се сумират.

Въз основа на изрази (7.9) и (7.10) е синтезирана логическата схема на полусуматора (фиг. 7.3.1). Като недостатък на тази схема се счита невъзможността да се минимизира функцията за сумата S_i и изискването за наличие на входните променливи x_i и y_i в права и инверсна форма. Този недостатък се избягва при схемата на полусуматор от фиг. 7.3.2 след преобразуване на израз (7.9) с цел представянето му в базис И-НЕ (7.11).



Фиг. 7.3.1. Полусуматор: а) схема в базис И-ИЛИ-НЕ; б) условно означение

$$S_i = \bar{x}_i y_i + x_i \bar{y}_i = (\bar{x}_i + \bar{y}_i) y_i + (\bar{x}_i + \bar{y}_i) x_i = x_i y_i y_i x_i x_i x_i. \quad (7.11)$$



Фиг. 7.3.2. Схема на полусуматор в базис И-НЕ

7.2.3.2. Едноразрядни пълни суматори

При събиране на многоразрядни числа към разрядите и се прибавя и преносът, получен от сумирането на двата по-младши разряда. Сумирането на три едноразрядни двоични числа се извършва по следния начин: $0+0+0=0$, $0+0+1=1$, $0+1+1=0$ и пренос 1 в старшия разряд, $1+1+1=1$ и пренос 1 в старшия разряд.

Комбинационна схема с три входа и два изхода реализираща тези правила се нарича едноразряден пълнен суматор. Пълният суматор извършва сумиране на кодовете на

Таблица 7.3-2.

Таблица на истинност на едноразряден пълнен суматор

x_i	y_i	P_{i-1}	P_i	S_i
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

три цифри, постъпващи на входовете му.

Таблица 7.3-2 е таблицата на истинност на едноразрядния пълнен суматор, от която могат да се получат каноничните форми на изходните логически функции S_i и P_i . За по-оптимална схема при синтеза, тези две функции се представят в

карти на Вейч и се търси възможност за тяхната минимизация.



От картите на Вейч е очевидно, че е възможна минимизация само на функцията P_i :

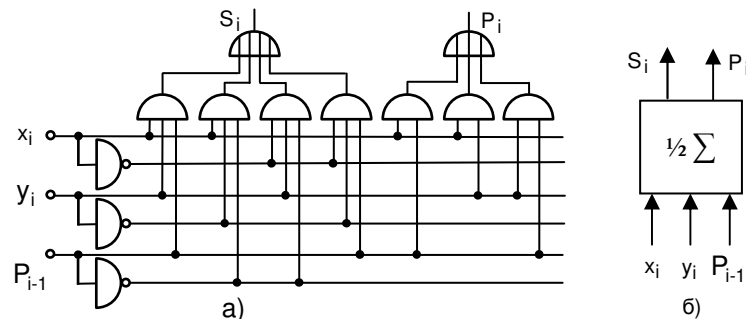
$$S_i = x_i y_i P_{i-1} + x_i \bar{y}_i \bar{P}_{i-1} + \bar{x}_i y_i \bar{P}_{i-1} + \bar{x}_i \bar{y}_i P_{i-1}; \quad (7.12)$$

$$P_i = x_i P_{i-1} + x_i y_i + y_i P_{i-1}. \quad (7.13)$$

Синтезираната логическа схема на едноразрядния пълнен суматор е показана на фиг. 7.3.3. За целта са използвани аналитичните изрази на получените функции (7.12) и (7.13) за осъществяване на схемата.

Предимство на схемата от фиг. 7.3.3 е нейната двустъпалност (без да се отчитат входните инвертори), но функцията S_i се реализира сложно. Това може да се избегне, ако при представянето на S_i се използва функцията P_i

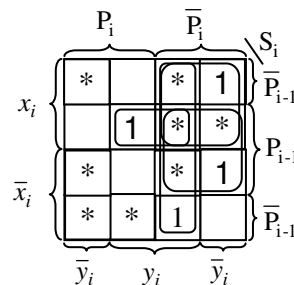
като базова функция (таблица 7.3-3 и получената чрез този метод схема, показана на фиг. 7.3.4).



Фиг. 7.3.3. Едноразряден пълнен суматор: а) схема в базис И-ИЛИ-НЕ; б) условно означение

Таблица 7.3-3

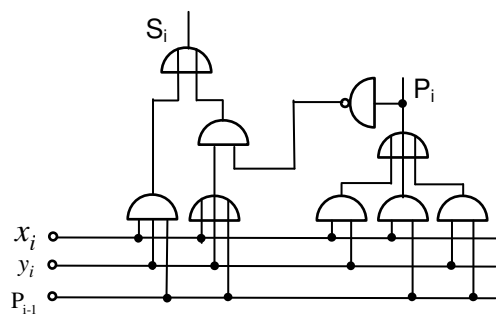
N_0	P_i	x_i	y_i	P_{i-1}	S_i
0	0	0	0	0	0
1	0	0	0	1	1
2	0	0	1	0	1
3	0	0	1	1	*
4	0	1	0	0	1
5	0	1	0	1	*
6	0	1	1	0	*
7	0	1	1	1	*
8	1	0	0	0	*
9	1	0	0	1	*
10	1	0	1	0	*
11	1	0	1	1	0
12	1	1	0	0	*
13	1	1	0	1	0
14	1	1	1	0	0
15	1	1	1	1	1



Функцията S_i се нанася в карта на Вейч и се намира нейната минимална форма (7.14).

$$S_i = x_i y_i P_{i-1} + \bar{P}_i x_i + \bar{P}_i y_i + \bar{P}_i P_{i-1} = x_i y_i P_{i-1} + \bar{P}_i (x_i + y_i + P_{i-1}) \quad (7.14)$$

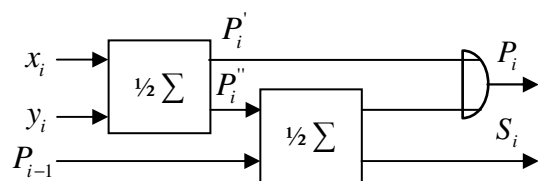
Функцията P_i не се променя и остава функция на три аргумента (7.13).



Фиг. 7.3.4. Едноразряден пълен суматор

Като резултат от такова представяне на функцията P_i като базова, се получава по-проста схема (фиг. 7.3.4). Входните променливи се използват само в права форма, но S_i се реализира чрез 5-стъпална схема. Винаги при синтез на схеми чрез използване на базова функция се увеличава тяхната стъпалност.

Чрез използването на два полусуматора и допълнителна логика се получава схема на пълен суматор (фиг. 7.3.5).



Фиг. 7.3.5. Пълен суматор от два полусуматора

7.2.4. Суматори в интегрално изпълнение

В TTL интегрално изпълнение се произвежда 4 битов пълен суматор от типа SN7483.