

## Использование особенностей ПЛИС для оптимизации схемы устройства управления

Баркалов А.А., Зеленёва И.Я., Лаврик А.С.  
University of Zielona Gora (Poland), ДонНТУ (Украина)  
A.Barkalov@iie.uz.zgora.pl

### Abstract

*Barkalov A.A., Zelenyova I.Y., Lavrik A.S. Application of PLD features for optimization of the control unit logical circuit. The method of hardware reduction is proposed oriented on compositional microprogram control units and CPLD chips. The method is based on a wide fan-in of PAL macrocells allowing using more than one source of microinstruction address. Such approach permits to minimize the number of PAL macrocells used for transformation of microinstruction address. Conditions for this method application and example of its application are given.*

### Введение

Одним из важных блоков цифровых систем является устройство управления. Если реализуемый алгоритм имеет линейный характер [1], то для его интерпретации может быть использована модель композиционного микропрограммного устройства управления (КМУУ) [1]. В настоящее время для реализации схем устройств управления широко используются программируемые логические интегральные схемы (ПЛИС), состоящие из макроячеек программируемой матричной логики (ПМЛ) [6, 7]. Высокая стоимость этого базиса требует решения актуальной задачи уменьшения числа микросхем ПЛИС в схеме КМУУ. Один из путей решения этой задачи – уменьшение числа термов в дизъюнктивных нормальных формах (ДНФ) реализуемых функций [2, 3]. В настоящей работе предлагается возможный подход к решению этой задачи, основанный на большом коэффициенте объединения по входу (порядка нескольких десятков) макроячеек ПМЛ [6, 7]. Разработанный метод ориентирован на КМУУ с преобразователем адреса микрокоманд [1], при этом алгоритм управления представлен в виде граф-схемы алгоритма (ГСА) [4].

Целью исследования является оптимизация комбинационной схемы КМУУ за счёт использования нескольких источников кода классов псевдоэквивалентных операторных линейных цепей (ОЛЦ). Задачей исследования является разработка метода синтеза, позволяющего уменьшить число макроячеек ПМЛ в схеме преобразователя адреса микрокоманды.

### Особенности КМУУ с преобразователем адреса микрокоманд

Пусть ГСА  $\Gamma$  представлена множествами вершин  $V$  и дуг  $E$ , соединяющих эти вершины. При этом  $V = \{b_0, b_E\} \cup E_1 \cup E_2$ , где  $b_0$  -

начальная вершина ГСА,  $b_E$  - конечная вершина ГСА,  $E_1$  - множество операторных вершин, где  $|E_1| = M$ ,  $E_2$  - множество условных вершин. В вершинах  $b_q \in E_1$  записаны наборы микроопераций  $Y(b_q) \subseteq Y$ , где  $Y = \{y_1, \dots, y_N\}$  - множество микроопераций. В вершинах  $b_q \in E_2$ , записаны элементы множества логических условий  $X = \{x_1, \dots, x_L\}$ . Пусть ГСА  $\Gamma$  является линейной, то есть включает более 75% операторных вершин.

Сформируем множество ОЛЦ  $C = \{\alpha_1, \dots, \alpha_G\}$  ГСА  $\Gamma$ , где каждая из ОЛЦ является последовательностью операторных вершин и каждой паре её соседних компонент  $b_i, b_j$  соответствует дуга  $\langle b_i, b_j \rangle \in E$ . Каждая ОЛЦ имеет только один выход  $O_g$  и произвольное число входов  $(g = 1, \dots, G)$ . Формальные определения ОЛЦ, их входов и выходов можно найти в [1]. Каждая вершина  $b_q \in E_1$  соответствует микрокоманде  $MI_q$ , хранимой в управляющей памяти (УП) КМУУ по адресу  $A(b_q)$ . Для адресации микрокоманд достаточно

$$R = \lceil \log_2 M \rceil \quad (1)$$

бит, представленных переменными  $T_r \in T$ , где  $|T| = R$ . Пусть ОЛЦ  $\alpha_g \in C$  включает  $F_g$  компонент, а адресация микрокоманд выполнена так, что

$$A(b_{gi+1}) = A(b_{gi}) + 1, \quad (2)$$

где  $b_{gi}$  -  $i$ -я компонента ОЛЦ  $\alpha_g \in C$ ,  $i = 1, \dots, F_g - 1$ .

Если выходы  $O_i, O_j$  соединены с входом одной и той же вершины ГСА  $\Gamma$ , то ОЛЦ  $\alpha_i, \alpha_j \in C$  являются псевдоэквивалентными ОЛЦ (ПОЛЦ) [1]. Найдём разбиение  $\Pi_C = \{B_1, \dots, B_l\}$  множества  $C_1 \subseteq C$  на классы ПОЛЦ. При этом ОЛЦ  $\alpha_g \in C_1$ , если её вход не связан с вершиной  $b_E$ , то есть  $\langle O_g, B_E \rangle \notin E$ . Закодируем классы  $B_i \in \Pi_C$  двоичными кодами  $K(B_i)$  разрядности

$$R_1 = \lceil \log_2 I \rceil \quad (3)$$

и используем для кодирования элементы множества  $J$ , где  $|J| = R_1$ . В этом случае для интерпретации ГСА  $\Gamma$  может быть использовано КМУУ с преобразователем адреса (Рис. 1), обозначаемое в дальнейшем символом  $U_1$  [1].

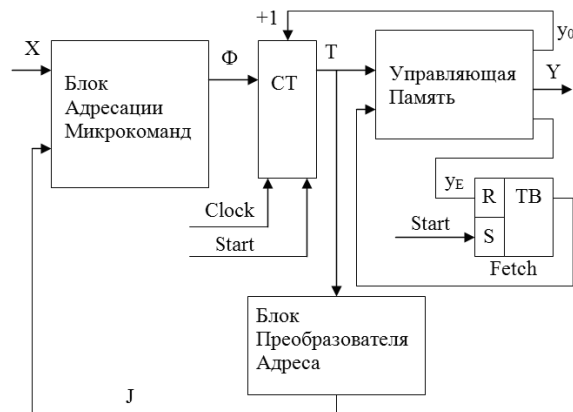


Рисунок 1 – Структурная схема КМУУ  $U_1$

По сигналу start в счётчик (СТ) записывается начальный адрес микропрограммы, а триггер выборки (ТВ) устанавливается в единичное состояние. При этом Fetch = 1, что разрешает выборку микрокоманд из УП. Если считанная микрокоманда  $MI_q$  не соответствует выходу ОЛЦ  $\alpha_g \in C_1$ , то одновременно с микрооперациями  $Y(b_q)$  формируется переменная  $y_0$ . Если  $y_0 = 1$ , то содержимое СТ увеличивается на 1, что соответствует режиму безусловного перехода (2) в пределах ОЛЦ. В противном случае  $y_0 = 0$  и блок адресации микрокоманд (БАМ) формирует функции

$$\Phi = \Phi(J, X) \quad (4)$$

для записи в T адреса входа очередной ОЛЦ. При этом блок преобразователя адреса (БПА) формирует функции

$$J = J(T), \quad (5)$$

равные единице в коде  $K(B_i)$ , где  $\alpha_g \in B_i$ . Если достигнут выход ОЛЦ  $\alpha_g \notin C_1$ , то формируется  $y_E = 1$ , триггер ТВ обнуляется и выборка микрокоманд прекращается.

Такая организация КМУУ позволяет уменьшить число термов в системе функций  $\Phi$  от  $N_1$  до  $N_0$ , где  $N_1$  – число строк таблицы переходов автомата Мура, а  $N_0$  – число строк таблицы переходов эквивалентного автомата Мили. Однако, схема БПА потребляет некоторые ресурсы ПЛИС или ППЗУ, из которых строится УП. В настоящей работе предлагается метод синтеза КМУУ  $U_2$ , в котором число термов в системе  $\Phi$   $N_2 = N_1$ , а схема БПА требует меньше аппаратных затрат, чем в КМУУ  $U_1$ . При определённых условиях этот блок может вообще исчезнуть.

### Основная идея предлагаемого метода

Отметим, что схемы блоков БАМ, БПА, СТ и ТВ реализуются в составе ПЛИС, а для реализации УП требуется ППЗУ, имеющие  $t$  выходов ( $t = 1, 2, 4, 8, 16$ ). Выполним адресацию ОЛЦ  $\alpha_g \in C_1$  таким образом, чтобы выполнялось равенство (2) и максимально возможное число классов  $B_i \in \Pi_C$  было выражено одним обобщённым интервалом R-мерного булева пространства. Для такой адресации необходимо разработать соответствующий алгоритм.

Пусть  $\Pi_C = \Pi_A \cup \Pi_B$ , где  $B_i \in \Pi_A$ , если этому классу соответствует один интервал, и  $B_i \in \Pi_B$  в противном случае. Источником кодов для классов  $B_i \in \Pi_A$  является счётчик СТ. Если выполняется условие

$$\Pi_B = \emptyset, \quad (6)$$

то блок БПА отсутствует. В противном случае преобразованию подлежат только адреса выходов ОЛЦ, входящих в классы  $B_i \in \Pi_B$ . Для кодирования этих классов достаточно

$$R_2 = \lceil \log_2 (I_B + 1) \rceil \quad (7)$$

переменных, где  $I_B = |\Pi_B|$ , единица прибавляется для кодирования ситуации  $B_i \in \Pi_A$ . Отметим, что часть кодов может быть реализована на свободных выходах ППЗУ. Пусть для кодирования микроопераций используется стратегия унитарного кодирования [1], тогда слово УП имеет  $N+2$  разрядов. Для реализации УП требуется

$$R_0 = \left\lceil \frac{N+2}{t} \right\rceil \quad (8)$$

микросхем с числом ячеек, не меньшим  $M$ . При этом

$$R_3 = R_0 * t - N - 2 \quad (9)$$

выходов ППЗУ являются свободными. Если

$$R_3 \geq R_2, \quad (10)$$

то источниками кодов классов  $B_i \in \Pi_B$  является УП и блок БПА отсутствует. В противном случае множество  $\Pi_B$  представляется в виде  $\Pi_E \cup \Pi_D$ , где  $I_E = |\Pi_E|, I_D = |\Pi_D|$ . При этом

$$I_E = 2^{R_3} - 1, \quad (11)$$

$$R_4 = \lceil \log_2(I_D + 1) \rceil. \quad (12)$$

Единица в (11) вычитается, чтобы отразить ситуацию  $B_i \notin \Pi_E$ . Единица в (12) прибавляется, чтобы отразить ситуацию  $B_i \notin \Pi_D$ . Итак, преобразованию подлежат только адреса выходов ОЛЦ  $\alpha_g \in B_i$ , где  $B_i \in \Pi_D$ . Если все множества  $\Pi_A$ ,  $\Pi_E$  и  $\Pi_D$  не являются пустыми, то для интерпретации ГСА Г предлагается КМУУ  $U_2$  (Рис. 2).

В КМУУ  $U_2$  коды  $K_A(B_i)$  представляются переменными  $T_r \in T$ , коды  $K_E(B_i)$  - переменными  $V_r \in V$ , где  $|V| = R_3$ , и коды  $K_D(B_i)$  - переменными  $Z_r \in Z$ , где  $|Z| = R_4$ . Отличие КМУУ  $U_2$  от КМУУ  $U_1$  в том, что БПА реализует функции

$$\Phi = \Phi(T, Z, V, X), \quad (13)$$

а БПК реализует функции

$$Z = Z(T). \quad (14)$$

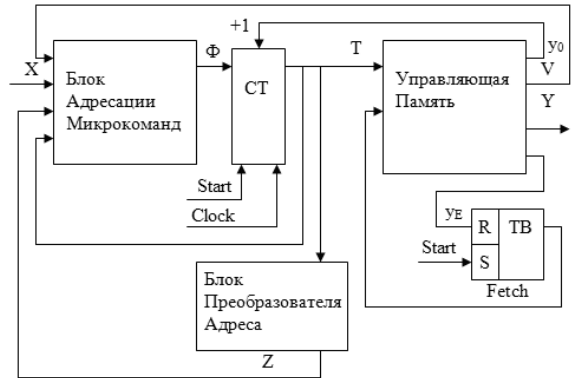


Рисунок 2 – Структурная схема КМУУ  $U_2$

Обозначим символом  $U_i(\Gamma_j)$  тот факт, что КМУУ  $U_i$  интерпретирует ГСА  $\Gamma_j$ , а  $Q_i(\Gamma_j)$  - число макроячеек ПМЛ в схеме БПА КМУУ  $U_i(\Gamma_j)$ , где  $i = 1, 2$ . Пусть на входы  $q$ -й макроячейки схемы БПА КМУУ  $U_i(\Gamma_j)$  поступает  $L_q$  логических условий, а каждая макроячейка имеет  $S$  входов. Применение предложенного метода имеет смысл, если выполняется условие

$$L_q + R + R_3 + R_4 \leq S, \quad (15)$$

где  $q = 1, \dots, Q_1(\Gamma_j)$ .

В настоящей работе предлагается метод синтеза КМУУ  $U_2$ , включающий следующие этапы:

1. Формирование множеств  $C, C_1$  и  $\Pi_C$  для ГСА Г.
2. Адресация микрокоманд.
3. Формирование множеств  $\Pi_A, \Pi_E$  и  $\Pi_D$ .
4. Кодирование классов  $B_i \in \Pi_E \cup \Pi_D$ .
5. Формирование содержимого управляющей памяти.
6. Формирование таблицы переходов КМУУ.
7. Формирование таблицы блока преобразователя адресов.
8. Синтез логической схемы КМУУ.

**Пример применения предложенного метода**

Пусть для некоторой ГСА  $\Gamma_1$  построены множества  $C = \{\alpha_1, \dots, \alpha_9\}$ ,  $C_1 = \{\alpha_1, \dots, \alpha_8\}$  и

$$\begin{aligned} \Pi_C &= \{B_1, \dots, B_5\}, & \text{где} & & \alpha_1 &= \langle b_1, b_2 \rangle, \\ \alpha_2 &= \langle b_3, \dots, b_6 \rangle, & & & \alpha_3 &= \langle b_7, b_8 \rangle, \\ \alpha_4 &= \langle b_5, \dots, b_{13} \rangle, & & & \alpha_5 &= \langle b_4, \dots, b_{17} \rangle, \\ \alpha_6 &= \langle b_{18}, \dots, b_{21} \rangle, & & & \alpha_7 &= \langle b_{22}, \dots, b_{25} \rangle, \\ \alpha_8 &= \langle b_{26}, \dots, b_{28} \rangle, & & & \alpha_9 &= \langle b_{29}, \dots, b_{31} \rangle, \\ B_1 &= \{\alpha_1\}, & B_2 &= \{\alpha_2, \alpha_3\}, & B_3 &= \{\alpha_4, \alpha_5\}, \\ B_4 &= \{\alpha_6, \alpha_7\}, & B_5 &= \{\alpha_8\}. & & \end{aligned}$$

Итак  $I = 5, R_1 = 3, J = \{J_1, J_2, J_3\}, M = 31, R = 5.$

Выполним адресацию микрокоманд, модифицировав алгоритм [1]. При этом  $A(b_1) = 00000, \dots, A(b_{25}) = 110000$   
 $A(b_{26}) = 11100, \dots, A(b_{28}) = 11110$   
 $A(b_{29}) = 11001, \dots, A(b_{31}) = 11011.$  Построим карту Карно, отмеченную переменными  $T_r \in T = \{T_1, \dots, T_5\}$ , в которой показаны выходы  $O_g$  ОЛЦ  $\alpha_g \in C$  и интервалы кодового пространства соответствующие классам  $B_i \in \Pi_C$  (Рис. 3).

	$T_1 T_2 T_3$							
$T_4 T_5$	000	001	011	010	110	111	101	100
00	*	*	$O_4$	*	$O_7$	*	$O_6$	$O_5$
01	$O_1$	$O_2$	*	*	*	*	*	*
11	*	$O_3$	*	*	*	*	*	*
10	*	*	*	*	$O_9$	$O_8$	*	*

Рисунок 3 – Карта Карно для выходов ОЛЦ КМУУ  $U_2(\Gamma_1)$

В этой карте символом \* обозначена ситуация, когда вершина  $b_q \in E_1$  с адресом  $A(b_q)$  не является выходом ОЛЦ  $\alpha_g \in C$ . Из рис. 3 можно найти кодовые интервалы, соответствующие выходам ОЛЦ классов  $B_i \in \Pi_C$ : классу  $B_1$  соответствует интервал  $0000*$ , классу  $B_2$  – интервал  $001**$ , классу  $B_3$  – интервалы  $01***$  и  $10000$ , классу  $B_4$  – интервалы  $101**$  и  $11000$ , классу  $B_5$  – интервал  $111**$ . Напомним, что класс  $B_6 = \{\alpha_9\}$  не рассматривается, так как ОЛЦ  $\alpha_9 \notin C_1$ .

Полученные интервалы определяют множества  $\Pi_A = \{B_1, B_2, B_3\}$  и  $\Pi_B = \{B_3, B_4\}$ . Пусть для ГСА  $\Gamma_1$   $N = 13$ , а для реализации УП используются ППЗУ с  $t = 4$ . В этом случае, из (9)

имеем  $R_3 = 1$ , а из (7) –  $R_2 = 2$ . Следовательно, условие (10) нарушается и в КМУУ  $U_2(\Gamma_1)$  используется БПК. Пусть  $\Pi_E = \{B_3\}$ , тогда  $\Pi_D = \{B_4\}$ . Итак, множества  $\Pi_A, \Pi_E, \Pi_D$  сформированы.

Очевидно,  $V = \{v_1\}, Z = \{z_1\}$ , пусть  $K_E(B_3) = 1, K_D(B_4) = 1$ , из предыдущего можно определить  $K_A(B_1) = 0000*, K_A(B_2) = 001**, K_A(B_5) = 111**.$

Содержание управляющей памяти формируется тривиально [1], этот этап здесь не рассматривается. Отметим только, что ячейки с адресами 10100 и 11000 кроме микроопераций содержат переменную  $v_1 = 1$ .

Пусть переходы для классов  $B_2, B_3, B_4$ , задаются следующей системой обобщённых формул перехода [4]:

$$\begin{aligned} B_2 &\rightarrow x_3 b_9 \vee \overline{x_3} b_{26}; \\ B_3 &\rightarrow x_1 b_{18} \vee \overline{x_1} x_2 b_{20} \vee \overline{x_1} \overline{x_2} b_{26}; \\ B_4 &\rightarrow x_5 b_{27} \vee \overline{x_5} b_5. \end{aligned} \quad (16)$$

Системе (16) соответствует фрагмент таблицы переходов КМУУ  $U_2(\Gamma_1)$ , имеющий 7 строк (Табл. 1).

Таблица 1. Фрагмент таблицы переходов КМУУ  $U_2(\Gamma_1)$

$B_i$	$K_A(B_i)$	$K_E(B_i)$	$K_D(B_i)$	$b_q$	$A(b_q)$	$X_h$	$\Phi_h$	$h$
	$T_1 T_2 T_3$ $T_4 T_5$	$v_1$	$z_1$					
$B_2$	001**	0	0	$b_9$	01000	$x_3$	$D_2$	1
				$b_{26}$	11100	$\overline{x_3}$	$D_1 D_2 D_3$	2
$B_3$	*****	1	0	$b_{18}$	10001	$x_1$	$D_1 D_5$	3
				$b_{20}$	10011	$\overline{x_1} x_2$	$D_1 D_4 D_5$	4
				$b_{26}$	11100	$\overline{x_1} \overline{x_2}$	$D_1 D_2 D_3$	5
$B_4$	*****	0	1	$b_{27}$	11101	$x_5$	$D_1 D_2 D_3 D_5$	6
				$b_5$	00100	$\overline{x_5}$	$D_3$	7

Связь этой таблицы с системой (16) очевидна. Отметим, что ситуация ( $v_1 = z_1 = 0$ ) соответствует классам  $B_i \in \Pi_A$ . В противном случае ( $v_1 \neq z_1 \neq 0$ ),  $B_i \in \Pi_E \cup \Pi_D$ , информация в столбце  $K_A(B_i)$  игнорируется, что отмечено символами «\*». Эта таблица является основой для формирования системы (13). Так из табл. 1, например, имеем части ДНФ:

$$D_1 = \overline{T_1 T_2 T_3} \overline{v_1 z_1 x_3} \vee \overline{v_1 z_1} \vee \overline{v_1 z_1 x_5};$$

$$D_2 = \overline{T_1 T_2 T_3} \overline{v_1 z_1} \vee \overline{v_1 z_1 x_1 x_2} \vee \overline{v_1 z_1 x_5}.$$

Таблица блока преобразователя адресов строится для классов  $B_i \in \Pi_D$ . В нашем примере она имеет 2 строки (Табл. 2).

Таблица 2. Таблица БПА устройства U<sub>2</sub>(Г<sub>1</sub>)

$\alpha_g$	$A(O_g)$	$B_i$	$K_D(B_i)$	$Z_j$	j
$\alpha_6$	10100	B <sub>4</sub>	1	z <sub>1</sub>	1
$\alpha_7$	1000				2

Из этой таблицы имеем систему (14), которая в нашем случае представляется следующей ДНФ:

$$z_1 = \overline{T_1 T_2 T_3 T_4 T_5} \vee \overline{T_1 T_2 T_3 T_4 T_5}.$$

Синтез логической схемы КМУУ U<sub>2</sub>(Г<sub>1</sub>) сводится к реализации систем (13) – (14) на ПЛИС. Эти задачи достаточно рассмотрены в литературе [1, 3] и здесь не рассматриваются.

Отметим, что в КМУУ U<sub>1</sub>(Г<sub>1</sub>) блок БПК задаётся таблицей, имеющей 8 строк, а число кодирующих переменных R<sub>1</sub> = 3. Пусть макроячейка ПМЛ имеет q = 3 терма, тогда для реализации БПК U<sub>2</sub>(Г<sub>1</sub>) необходима одна макроячейка. Если в U<sub>1</sub>(Г<sub>1</sub>) классы  $B_i \in \Pi_C$  закодированы тривиальным образом (K(B<sub>1</sub>) = 000, ..., K(B<sub>5</sub>) = 100), то для реализации БПК требуется 4 макроячейки ПМЛ с q = 3. При этом число макроячеек в схемах блоков БАП для КМУУ U<sub>1</sub>(Г<sub>1</sub>) и U<sub>2</sub>(Г<sub>1</sub>) совпадает.

### Заключение

Предлагаемый метод оптимизации схемы КМУУ с преобразователем адреса ориентирован на уменьшение числа макроячеек ПМЛ в схеме БПА. При этом число макроячеек в схеме формирования адреса и число микросхем ППЗУ в управляющей памяти остаётся неизменным. Метод основан на использовании трёх источников кодов классов псевдоэквивалентных ОЛЦ.

Научная новизна предложенного метода заключается в использовании особенностей базиса ПЛИС, а именно большого коэффициента объединений по входу, для уменьшения аппаратных затрат в схеме БПА. Отметим, что при выполнении условия (10) этот блок вообще отсутствует. Практическая значимость этого метода заключается в уменьшении числа микросхем при реализации схемы КМУУ, что позволяет получить схемы, обладающие меньшей стоимостью по сравнению с

известными из литературы аналогами. Рассмотренные нами примеры показали, что число макроячеек в блоке БПА уменьшается на 60-70%. При этом общее число макроячеек в схеме КМУУ U<sub>2</sub>(Г<sub>1</sub>) до 10% меньше, чем в КМУУ U<sub>1</sub>(Г<sub>1</sub>).

Для повышения эффективности метода необходимо разработать алгоритм адресации микрокоманд КМУУ, уменьшающий число ОЛЦ, адреса выходов которых должны преобразовываться. Это относится к дальнейшему направлению наших исследований, в качестве проверки возможности использования метода для базиса «систем-на-кристалле» [2], который имеет внутренние ресурсы для реализации, как произвольной логики, так и управляющей памяти КМУУ.

### Литература:

1. Баркалов А.А. Синтез устройств управления на программируемых логических устройствах. – Донецк: ДНТУ, 2002. – 262 с.
2. Грушвицкий Р.И., Мурсаев А.Х., Угрюмов Е.П. Проектирование систем с использованием микросхем программируемой логики. – СПб: БХВ. – Петербург, 2002. – 608 с.
3. Соловьев В.В. Проектирование цифровых схем на основе программируемых логических интегральных схем. – М.: Горячая линия-ТЕЛЕКОМ, 2001. – 636 с.
4. Baranov S. Logic Synthesis for Control Automata. – Kluwer Academic Publishers, 1994. – 312 pp.
5. DeMicheli G. Synthesis and Optimization of Digital Circuits. – McGraw-Hill, 1994. – 636 pp.
6. Altera devices overview.  
[http://www.altera.com/products/devices/commo/dev-family\\_overview.html](http://www.altera.com/products/devices/commo/dev-family_overview.html).
7. Xilinx CPLDs  
[http://www.xilinx.com/products/silicon\\_solutions/cplds/index.htm](http://www.xilinx.com/products/silicon_solutions/cplds/index.htm).