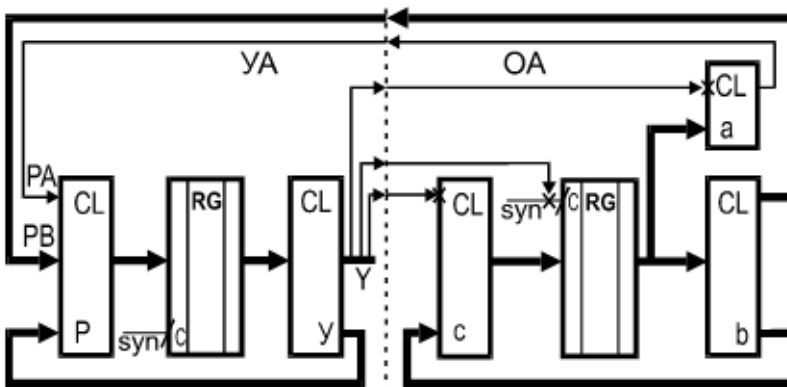


# Конвейерный способ построения взаимодействия операционного и управляющего автоматов для схем синхронных цифровых автоматов

Романов Александр Михайлович,  
доцент кафедры Вычислительной техники  
Московского технологического университета

## АННОТАЦИЯ

Рассматриваются способы построения взаимодействия операционных и управляющих автоматов для схем синхронных цифровых автоматов. Предлагается использование конвейерных вариантов с целью сокращения длительности такта взаимодействия.



При проектировании вычислительного устройства, выполняющего сложную обработку цифровой информации по заданному алгоритму, одним из вариантов декомпозиции является представление синхронного

Рис.1. Последовательный вариант взаимодействия вычислителя в виде композиции двух автоматов операционного и управляющего /1/.

Совместная работа этих двух автоматов может быть описана следующим образом. На границе такта (фронт, синхронизирующий регистры автоматов) изменяется содержимое регистров как OA так и UA и соответственно выходные значения автоматов. Это приводит к тому, что UA формирует новые значения сигналов для OA .

Последовательная схема взаимодействия автоматов представлена на рис.1.

Отметим, что сигналы признаков могут быть двух типов

- $PA(t) = f_A(Y(t))$  зависит без сдвига от сигналов управления,
- $PB(t+1) = f_B(Y(t))$  зависит со сдвигом от сигналов управления.

Минимальная длительность такта работы схемы определяется наиболее длинными цепями между регистрами. Для схемы на рис.1, которую будем называть последовательной схемой взаимодействия, зададимся (так чаще всего бывает), что такой критической цепью является цепь (CL<sub>y</sub>, CL<sub>a</sub>, CL<sub>p</sub>, RG). Поэтому длительность такта определяется:

$$T > t_y + t_a + t_p + t_{rg},$$

где  $t_j$  - время установления соответствующего компонента цепи.

Чтобы сократить длину этой цепи, можно применить другой вариант взаимодействия автоматов — *конвейерный* - рис.2. UA для этого варианта взаимодействия отличается от UA варианта последовательного взаимодействия добавленным регистром RG\_FF (регистром флагов) и несколько

иной компоновкой регистров и комбинационных схем. При этом варианте взаимодействия такой длинной цепи, как в предыдущем случае, нет. Эта цепь разделена регистрами RG\_FF и RG\_MK (регистр микрокоманды) на две цепи. Продолжительность такта становится меньше и ее можно определить следующим образом:

$$T > \max[ t_a , (t_p + t_y) ] + t_{rg} ,$$

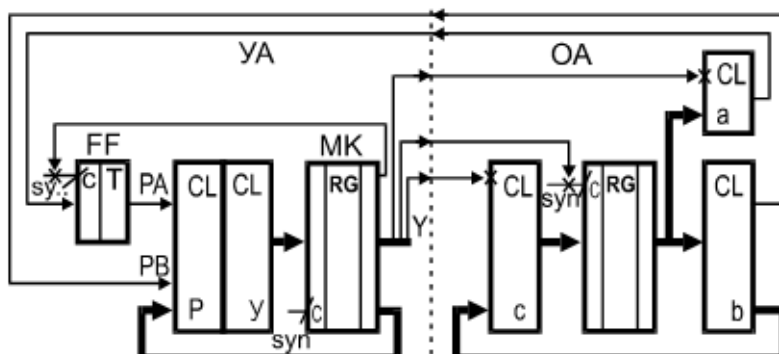


Рис. 2 . Конвейерный вариант

При конвейерном варианте взаимодействия  $PA(t+1)=f(Y(t))$ , Тогда фрагмент микропрограммы

$mS\{\dots; pA=f(\dots)\}$

«GO(pA;mi,mj)»,

выполняемый в последовательной схеме за один такт, в конвейерном варианте должен быть модифицирован следующим образом:

$mS\{\dots; pA=f(\dots)\}$

$mS'\{\text{нет операции}\}$

«GO(pA;mi,mj)»

Особенностью реально существующих структур является большая разрядность микрокоманды и малая разрядность (чаще всего 1) анализируемых на каждом шаге признаков. Поэтому является оправданной такая структуризация комбинационной части схемы управляющего автомата, в которой выделяется функция мультиплексирования одного признака на каждом шаге выполнения алгоритма и реализация остальной значительной части логики в виде ПЗУ.

В конвейерном варианте при выполнении условных переходов по флагам, зависящим без сдвига от сигналов управления, приходится добавлять еще один такт для того, чтобы «увидеть» значение переменной, по которой выполняется ветвление, и выбрать нужную микроинструкцию. Можно построить управляющий автомат, в котором для ускорения выполнения микропрограммы выполняются следующие действия. Предварительно выбирается из ПЗУ одна из двух альтернативных микроинструкций, соответствующая наиболее вероятному значению флага. Это значение должно храниться в той микроинструкции, после которой выполняется ветвление. В конце такта выработанное реальное значение переменной сравнивается с предсказанным, если они совпадают, то выбранная из ПЗУ микроинструкция записывается в выходной регистр, если нет, то предыдущий такт продлевается, т.е. не синхронизируются операционный автомат и RG'MCI.

Микропрограмма будет такой же, как и для последовательного варианта взаимодействия, но в микроинструкцию добавляются два разряда:

$f = \{ 1, \text{если используется предвосхищение}; 0, \text{если нет} \};$

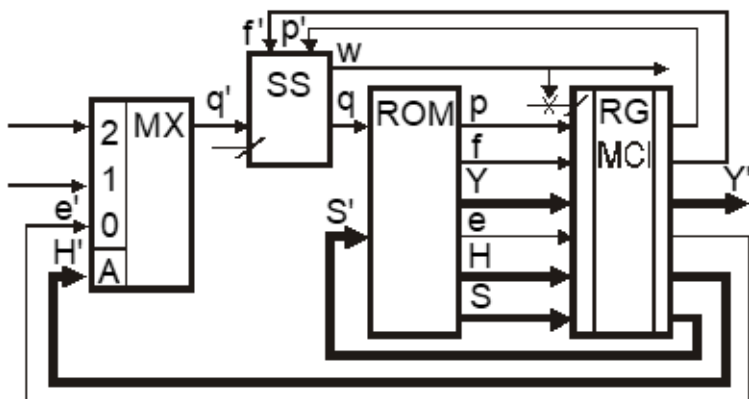


Рис.3. УА с предвосхищением

$p$  — наиболее вероятное значение переменной ветвления.

В схему включается автомат SS (рис.3), который вырабатывает два сигнала:

$q$ - разряд адреса ПЗУ;

$w$  — управляет условной синхронизацией операционного автомата и RG'MCI; эта переменная должна зависеть без сдвига от входных переменных автомата SS, т.к. установившееся значение  $w$  должно предшествовать рабочему фронту сигнала синхронизации.

Итак, SS — автомат, который может находиться в одном из двух состояний  $s_0$  и  $s_1$ :

состояние	$f'$	$p',q'$	$q$	$w$	переход
$s_0$	0	x	$q'$	1	$s_0$
$s_0$	1	$p'=q'$	$p'$	1	$s_0$
$s_0$	1	$p' \neq q'$	$p'$	0	$s_1$
$s_1$	x	x	$q'$	1	$s_0$

Аппаратура микропрограммного управляющего автомата несколько увеличилась при сокращении длительности такта. При этом основная доля прироста приходится на ПЗУ, увеличенное на два разряда в каждой ячейке.

#### Список литературы

1. Гилл А. Введение в теорию конечных автоматов. —М.: Наука, 1966 —272с.