

ИНФОРМАТИКА И ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА

УДК 681.5

С. М. Вертешев, В. А. Коневцов, И. А. Полетаев

УСТРОЙСТВА ДИСКРЕТНОЙ АВТОМАТИКИ ДЛЯ АСУ ТП

Статья представляет функциональные возможности создания устройств дискретной автоматики в комплексе программных средств проектирования систем цифрового управления (в комплексе ПСПСЦУ) для автоматизированных систем управления технологическими процессами (АСУ ТП). Оцениваются возможности комплекса ПСПСЦУ в реализации одноктактных и многотактных релейно-контактных схем в сравнении с требованиями международного стандарта МЭК 61131-3:2003 (Часть 3: Языки программирования).

Ключевые слова: компаратор векторов логических сигналов, бистабильный элемент, сдвигающий регистр, счетчик тактов, распознаватель фронта логического сигнала, функция голосования.

Введение

Для проектирования систем блокировок, защиты технологического оборудования от перегрузок, управления режимами технологического оборудования (включая режимы пуска, останова, переключений в нормальных условиях эксплуатации и в аварийных ситуациях) используются устройства дискретной автоматики. Эти устройства реализуются техническими средствами по стандарту МЭК 61131-1:2003 (Часть 1: Общие данные), программно-логическими контроллерами (ПЛК). Математическим аппаратом описания подобных устройств является аппарат исчисления высказываний — булева алгебра. Устройства логического управления классифицируются на комбинационные (не имеющие встроенную память — автоматы без памяти) и последовательностные (автоматы с памятью). В соответствии с этой классификацией различают устройства одноктактного (комбинационного) и многотактного (последовательностного) управления [2, 6]. Любая логическая переменная и функция двужначной логики определяется на множестве значений $\{0,1\}$ или {ложь, истина}. Функция двужначной логики может быть одноместной, двуместной и многоместной. К одноместным функциям относятся функции повторения $Y = X$, отрицания $Y = \bar{X}$, константа-ложь $Y = 0$, константа-истина $Y = 1$. Для двух независимых переменных можно получить 16 различных функций [2, 6], которые могут быть эквивалентно выражены через функцию отрицания (NOT) и функцию конъюнкции (AND) или через функцию отрицания (NOT) и функцию дизъюнкции (OR). Наряду с этими функциями используется функция антивалентности (XOR, неравнозначность, сложение по модулю два). Эти четыре функции определены в составе операций модулей вычисления выражений комплекса ПСПСЦУ [5]:

многместная конъюнкция AND

$$Y = X_1 \wedge X_2 \wedge \dots \wedge X_n = \begin{cases} 1 & \text{при } X_1 = X_2 = \dots = X_n = 1 ; \\ 0 & \text{иначе;} \end{cases}$$

многместная дизъюнкция OR

$$Y = X_1 \vee X_2 \vee \dots \vee X_n = \begin{cases} 0 & \text{при } X_1 = X_2 = \dots = X_n = 0 ; \\ 1 & \text{иначе;} \end{cases}$$

многместная неравнозначность XOR

$$Y = X_1 \oplus X_2 \oplus \dots \oplus X_n = \begin{cases} 0 & \text{при } \sum(X_1, X_2, \dots, X_n) = \text{четное} ; \\ 1 & \text{иначе;} \end{cases}$$

отрицание NOT

$$Y = \bar{X} = \begin{cases} 1 & \text{при } X = 0 \\ 0 & \text{при } X = 1 \end{cases}$$

Это базовые функции реализации устройств дискретной автоматики, осуществляющих запоминание, хранение, счет, анализ, сравнение логических сигналов и векторов логических сигналов. Такими устройствами являются триггеры, регистры, счетчики, дешифраторы, шифраторы, коммутаторы, компараторы схемотехники ЭВМ [2, 6] и т. д. Стандартизированные функциональные модули комплекса ПСПСЦУ определены на основе анализа математических методов теории управления [1, 3–5]. Сравнительная оценка свойств комплекса ПСПСЦУ и функциональных требований по стандарту МЭК 61131-3 дана в таблице 1.

Таблица 1

Оценка свойств комплекса ПСПСЦУ

№	Группы модулей дискретной автоматики	ПСПСЦУ	МЭК 61131-3
1	Компараторы векторов логических сигналов	+	–
2	Бистабильные элементы (RS, SR)	+	+
3	Сдвигающие регистры	+	–
4	Счетчики	+	+
5	Распознаватели фронта сигнала	+	+
6	Функции голосования	+	–
7	Дешифраторы	+	–
8	Шифраторы	+	–
9	Преобразователи битовой последовательности в вектор логических сигналов	+	–
10	Преобразователи вектора логических сигналов в битовую последовательность	+	–
11	Индексные регистры	+	–
12	Датчики времени (импульс, задержка включения, задержка выключения)	+	+
13	Коммутаторы сигналов	+	–
14	Генераторы сигналов	+	–

Примечание к таблице: (+) — наличие, (–) — отсутствие стандартизированной группы модулей

1. Компараторы векторов логических сигналов

Сравнение векторов логических сигналов (например, определение положения десятков и сотен дискретных исполнительных механизмов по состоянию конечных выключателей в АСУ ТП) производится с помощью компараторов [2]:

$$Y_i = \begin{cases} X_i & \text{при } f=0 & \text{повторение} \\ \overline{X_i} & \text{при } f=1 & \text{инверсия} \\ X_i \oplus Z_i & \text{при } f=2 & \text{антивалентность} \\ X_i \wedge Z_i & \text{при } f=3 & \text{конъюнкция} \\ X_i \vee Z_i & \text{при } f=4 & \text{дизъюнкция} \end{cases} \quad (1)$$

где $i = 1, 2, \dots, n$; n — размерность векторов сигналов, f — параметр выбора операции, X_i, Z_i — компоненты векторов размерности n .

2. Бистабильные элементы

Для хранения значений одного сигнала логического вида используются RS и SR триггеры [6], имеющие два устойчивых состояния при сбросе или установке (бистабильные элементы). Эти триггеры имеют вход сброса R, вход установки S, вход состояния в предыдущем такте, выход состояния и инверсный выход состояния. Для хранения вектора из n логических сигналов можно определить «пакетный» RS-триггер (такая упаковка практикуется в технике изготовления интегральных микросхем [7]). При программировании это реализуется в соответствии с выражением [6]:

$$\left. \begin{aligned} Q_i(k) &= \begin{cases} 0 & \text{при } R_i=1 \\ 1 & \text{при } S_i=1 \\ Q_i(k-1) & \text{иначе} \end{cases} \\ Y_i(k) &= \overline{Q_i(k)} \end{aligned} \right\} \quad (2)$$

где $i = 1, 2, \dots, n$; n — число триггеров в «пакете», а при $n = 1$ триггер становится обычным. В «пакет» логических сигналов включаются сигналы, обладающие близкой инерционностью, т. е. одинаковой частотой изменения. Если вектор логических сигналов должен содержать сигналы различной инерционности, то время отработки схемы, содержащей блок такого функционального модуля, определяется сигналом наибольшей частоты изменения его значений. На выходе состояния триггера с приоритетом сброса при $R = S = 1$ в одном и том же такте всегда значение сигнала равно нулю. На выходе состояния триггера с приоритетом установки при $R = S = 1$ в одном и том же такте всегда значение сигнала равно единице:

$$\left. \begin{aligned} Q_i(k) &= \begin{cases} 1 & \text{при } S_i=1 \\ 0 & \text{при } R_i=1 \\ Q_i(k-1) & \text{иначе} \end{cases} \\ Y_i(k) &= \overline{Q_i(k)} \end{aligned} \right\} \quad (3)$$

где $i = 1, 2, \dots, n$; n — число триггеров в «пакете».

В классическом определении RS и SR триггеров память на входе состояния в предыдущем такте скрыта, это внутренняя связь модуля, она разрушается при выключении электропитания. В определениях комплекса ПСПСЦУ [5] эта связь доступна инженеру-системотехнику, и память на входе состояния в предыдущем такте может быть сохранена при исчезновении электропитания ПЛК.

3. Сдвигающие регистры

Для хранения сигналов различного вида используются многоместные накопители («ячейки» памяти) — регистры различного действия: последовательный ввод-последовательный вывод, последовательный ввод-параллельный вывод, параллельный ввод-параллельный вывод, параллельный ввод-последовательный вывод. В системах с памятью (фиксация тренда предаварийной ситуации, корректирующие устройства высокого порядка, идентификация и имитация систем и т. д.) используются накопители с глубокой памятью. К таким накопителям относятся сдвигающие регистры [5]:

$$\left. \begin{aligned} Z_2 &= \begin{cases} 1 & \text{при } (q \geq N) \wedge (Z_0 = 0) \\ 0 & \text{при } (q < N) \vee (Z_0 = 1); \end{cases} \\ q &= \begin{cases} q_0 + 1 & \text{при } (q < N) \wedge (Z_1 = 1) \wedge (Z_0 = 0) \\ N & \text{при } (q \geq N) \wedge (Z_0 = 0) \\ 0 & \text{при } Z_0 = 1; \end{cases} \\ Y_k &= \begin{cases} X_k & \text{при } (Z_1 = 1) \wedge (Z_0 = 0) \\ X_{k-1} & \text{при } (Z_1 = 0) \wedge (Z_0 = 0) \\ 0 & \text{при } Z_0 = 1 \end{cases} \end{aligned} \right\}, \quad (4)$$

где $k = 1, 2, \dots, N$; N — длина тренда; q_0 — начальная длина тренда; Z_0 — сброс регистра; Z_1 — сдвиг регистра; q — текущая длина регистра; $X(k), X(k-1), \dots, X(k-N)$ — значения переменной в такты $(k-1), (k-2), \dots, (k-N)$; $y(k), Y(k-1), \dots, Y(k-N)$ — содержимое регистра.

4. Счетчики тактов

Счетчики различают по характеру счета (прямые, реверсивные), по представлению в системе счисления (двоичные, десятичные), по способу управления счетом (синхронные, асинхронные). Счетчик («пакетный») ведет относительный счет времени, т. е. счет в тактах схемы управления. В «пакет» счетчиков включаются сигналы, обладающие близкой инерционностью, т. е. одинаковой частотой изменения. Если к «пакету» счетчиков должны подключаться сигналы различной инерционности, то время отработки схемы, содержащей блоки такого функционального модуля, определяется сигналом наибольшей частоты изменения его значений:

$$\left. \begin{aligned} Z_i &= \begin{cases} 1 & \text{при } T_3_i = TT_i \\ 0 & \text{иначе} \end{cases} \\ Y_i &= \begin{cases} TH_i & \text{при } C_i = 1 \\ TT_i + 1 & \text{при } TT_i < T_3_i \\ TT_i - 1 & \text{при } TT_i > T_3_i \\ T_3_i & \text{при } TT_i = T_3_i \end{cases} \end{aligned} \right\}, \quad (5)$$

где C_i — сброс счетчика $i = 1, 2, \dots, n$; n — число счетчиков в пакете; TH_i — начальное значение счетчика i при сбросе; T_3_i — задание счетчика i ; TT_i — текущее значение счетчика i ; $Z_i = 1$ — счетчик i заполнен.

5. Распознаватели фронта сигнала

Анализ изменения логического сигнала (распознавание фронта) производится пороговыми логическими элементами, называемыми триггерами фронта сигнала.

Триггеры фронтов различают по характеру изменения значения логического сигнала:

- **триггер переднего фронта**

$$\left. \begin{aligned} Q_i &= \begin{cases} 1 & \text{при } (X_i(k-1)=0) \wedge (X_i(k)=1) \\ 0 & \text{иначе;} \end{cases} \\ Y_i &= X_i(k); \\ q_i &= \overline{Q_i}, \\ & \text{где } i=1,2,\dots,n \end{aligned} \right\}, \quad (6)$$

где n — число триггеров в «пакете»; $X_i(k)$ — значение анализируемого сигнала в текущем такте кТ; $X_i(k-1)$ — значение анализируемого сигнала в предыдущем такте $(k-1)T$; Y_i — память анализируемого сигнала; q_i — отсутствие переднего фронта;

- **триггер заднего фронта:**

$$\left. \begin{aligned} Q_i &= \begin{cases} 1 & \text{при } (X_i(k-1)=1) \wedge (X_i(k)=0) \\ 0 & \text{иначе;} \end{cases} \\ Y_i &= X_i(k) \\ q_i &= \overline{Q_i}, \\ & \text{где } i=1,2,\dots,n \end{aligned} \right\}, \quad (7)$$

где n — число триггеров в «пакете»; $X_i(k)$ — значение анализируемого сигнала в текущем такте кТ; $X_i(k-1)$ — значение анализируемого сигнала в предыдущем такте $(k-1)T$; Y_i — память анализируемого сигнала; q_i — отсутствие заднего фронта;

- **триггер (любого) фронта:**

$$\left. \begin{aligned} Q_i &= \begin{cases} 1 & \text{при } X_i(k-1) \neq X_i(k) \\ 0 & \text{иначе} \end{cases} \\ Y_i &= X_i(k) \\ q_i &= \overline{Q_i} \\ & \text{где } i=1,2,\dots,n \end{aligned} \right\} \quad (8).$$

где n — число триггеров в «пакете»; $X_i(k)$ — значение анализируемого сигнала в текущем такте кТ; $X_i(k-1)$ — значение анализируемого сигнала в предыдущем такте $(k-1)T$; Y_i — память анализируемого сигнала; q_i — отсутствие фронта.

В «пакет» логических сигналов также включаются сигналы, обладающие близкой инерционностью, т.е. одинаковой частотой изменения. Если же вектор ло-

гических сигналов должен содержать сигналы различной инерционности, то время обработки схемы, содержащей блок такого функционального модуля, определяется сигналом наибольшей частоты изменения его значений. Определение триггеров фронта дается также в «пакетной» форме. В момент изменения сигнала на выходах триггеров соответствующие логические значения сигналов возникают на время, равное одному такту обработки схемы, если частота изменения сигнала на соответствующих входах не больше частоты запуска схемы управления.

6. Функции голосования

Обработка сигналов логического вида как аргументов многоместных функций может выполняться с помощью так называемых функций голосования в схемах распознавания и исправления ошибок при передаче информации:

$$Y = \begin{cases} 1 & \text{при } \sum(X_1, X_2, \dots, X_n) \geq m, \\ 0 & \text{иначе,} \end{cases} \quad (9)$$

где n — число логических переменных, m — кворум системы голосования.

Зависимость (9) определяет функцию голосования конъюнктивно (мажоритарный элемент [6]) для любой комбинации логических сигналов из n по m . При $n=m$ выражение (9) соответствует многоместной конъюнкции, при $m = 1$ выражение (9) соответствует многоместной дизъюнкции, при $n = m = 1$ выражение (9) соответствует функции повторения.

7. Дешифраторы и шифраторы

Существует широкое многообразие преобразователей кодов. Особую роль в проектировании последовательностных схем, конечных автоматов, цифровых вычислительных устройств играют шифраторы и дешифраторы (эти модули также отсутствуют в качестве стандартных по МЭК 61131-3). Число состояний конечного автомата соответствует числу входов шифратора и числу выходов дешифратора [2,6]. Функция дешифратора состоит в преобразовании любого числа X системы счисления с основанием B из N позиций, причем $B^N > 0$, в вектор логических сигналов размерностью $m = B^N$. Сигнал на выходе дешифратора Y_j с номером $j = X$ всегда равен логической единице, причем $1 \leq j \leq m$, а значения сигналов на других выходах Y_j с номерами $j \neq X$ всегда равны логическому нулю:

$$Y_j = \begin{cases} 1 & \text{при } j = X \\ 0 & \text{для всех } j \neq X, \end{cases} \quad (10)$$

где $j=1,2,\dots,m$, $1 \leq X \leq m$.

Шифратор выполняет обратное действие к действию дешифратора (10), т. е. преобразует логический входной вектор X_1, X_2, \dots, X_m размерности m в номер j того входа X_j , который имеет логическое значение, равное 1. Все входы шифратора, кроме единственного входа (который обозначает текущий номер состояния объекта), должны быть с нулевыми значениями сигналов:

$$Y = \begin{cases} 1 & \text{при } n \neq t \\ 0 & \text{иначе,} \end{cases} \quad (11)$$

$$m = \begin{cases} i & \text{для } X_i = 1 \text{ и всех } X_j = 0, j \neq i \\ n & \text{иначе,} \end{cases} \quad (12)$$

8. Преобразователи битовой последовательности в вектор логических сигналов и обратно

При вводе дискретных сигналов двоичные сигналы группируются чаще всего побайтно, как натуральные числа, которые в дальнейшем используются как логические сигналы. Для получения отдельных, не сгруппированных в байты, логических сигналов используется модуль преобразования последовательности натуральных чисел в поразрядный позиционный код, каждый разряд которого занимает один байт.

При выводе дискретных сигналов с выходов комбинационных или последовательностных схем необходимо преобразование вектора логических сигналов в сигналы, упакованные в байты. Для этого используется модуль преобразования вектора логических сигналов в последовательность натуральных чисел.

9. Индексные регистры

Если сигналы вектора X_1, X_2, \dots, X_n могут принимать произвольные логические значения, то входы с логическими единицами можно «пометить», т. е. проиндексировать зависимостью (регистр индексации):

$$\left. \begin{aligned} q &= \sum (X_1, X_2, \dots, X_n) \\ Y_j &= i \cdot X_i \quad \text{для всех } X_i \neq 0, \end{aligned} \right\} \quad (13)$$

где q — число входов с единичным значением сигнала $X_i, j=1, 2, \dots, q$.

Регистр индексации представляет собой функцию кодирования с переменным числом выходов. Этот модуль может быть использован, например, для нумерации входов, имеющих единичные значения сигналов (например, номеров исполнительных механизмов, состояния которых отклоняются от нормы). Если значения сигналов на всех n входах единичные, то зависимость (13) даст следующие значения выходов: $q=n, Y_1=1, Y_2=2, \dots, Y_n=n$. Если значения сигналов на всех входах нулевые, то $q=0$, а выходы Y_j не существуют (ничто).

10. Датчики времени

Различают таймеры (датчики времени), формирующие импульс функциональным блоком TP, задержку включения функциональным блоком TON, задержку выключения функциональным блоком TOF. Любой из этих функциональных блоков имеет входы IN и PT типов BOOL и TIME соответственно и выходы Q и ET типов BOOL и TIME соответственно, причем IN-сигнал запуска блока; PT-вход задания времени; Q-наличие импульса; ET-текущее время, счетчик относительного времени.

Таймер TP(IN, PT, Q, ET): если IN = FALSE, то Q = FALSE и ET = 0. При переднем фронте сигнала на входе, IN = TRUE, выход Q принимает значение TRUE и

счетчик ET начинает отсчет времени до значения PT по тактам. Таким образом, на выходе Q формируется импульс длительностью PT по переднему фронту сигнала на входе IN:

$$ET = \begin{cases} 0 & \text{if } IN = FALSE \\ ET + 1 & \text{if } (ET < PT) \& (IN = TRUE), \end{cases} \quad (14)$$

$$Q = \begin{cases} FALSE & \text{if } (IN = FALSE) \\ TRUE & \text{if } (IN = TRUE). \end{cases} \quad (15)$$

Таймер с задержкой включения TON(IN, PT, Q, ET): если IN = FALSE, то Q = FALSE и ET = 0. При переднем фронте сигнала на входе, IN = TRUE, выход Q принимает значение TRUE и счетчик ET начинает отсчет времени до значения PT по тактам. Сигнал на выходе Q = TRUE, если IN = TRUE и ET = PT, иначе Q = FALSE. Таким образом, на выходе Q формируется импульс с задержкой PT от переднего фронта сигнала на входе IN:

$$ET = \begin{cases} 0 & \text{if } (IN = FALSE) \\ ET + 1 & \text{if } (ET < PT) \& (IN = TRUE), \end{cases} \quad (16)$$

$$Q = \begin{cases} FALSE & \text{if } (IN = FALSE) \\ TRUE & \text{if } (IN = TRUE) \& (ET = PT). \end{cases} \quad (17)$$

Таймер с задержкой выключения TOF(IN, PT, Q, ET): если IN = TRUE, то Q = TRUE и ET = 0. При заднем фронте сигнала на входе, IN = FALSE, начинается отсчет времени на выходе ET. Выход Q = FALSE, если IN = FALSE и ET = PT, иначе Q = TRUE. Таким образом, выход Q сбрасывается с задержкой PT от заднего фронта сигнала на входе IN:

$$ET = \begin{cases} 0 & \text{if } (IN = TRUE) \\ ET + 1 & \text{if } (ET < PT) \& (IN = FALSE). \end{cases} \quad (18)$$

$$Q = \begin{cases} TRUE & \text{if } (IN = TRUE) \\ FALSE & \text{if } (IN = FALSE) \& (ET = PT). \end{cases} \quad (19)$$

11. Коммутаторы сигналов

Все переменные связи в системе управления осуществляются с помощью коммутаторов. Различаются коммутаторы [6] входных сигналов (мультиплексоры или коммутаторы чтения), коммутаторы выходных сигналов (демультиплексоры или распределители или коммутаторы записи) и матричные коммутаторы (мультиплексоры-демультиплексоры или коммутаторы чтения-записи)[2].

$$Y_i = (AC + HЭ + \{HB - 1\} KЭ + i) \text{ при чтении } m = 1 \quad (20)$$

$$(AC + HЭ + \{HB - 1\} KЭ + i) = X_i \text{ при записи } m = 0 \quad (21)$$

где $i = 1, 2, \dots, KЭ$, чтение (1) запись (0), m -команда чтения/записи (.) — содержимое по адресу AC; HЭ — номер первого входа-выхода; HB — номер коммутируемого вектора; KЭ — число коммутируемых значений.

Функция мультиплексора получается при подстановке $m = 1$ в выражение (20), а функция демультиплексора — при подстановке $m = 0$ в выражение (21). Каждая

из переключаемых величин $X_1, X_2, \dots, X_{k\alpha}, Y_1, Y_2, \dots, Y_{k\alpha}$ может быть отдельным значением сигнала (скаляром) или вектором значений сигналов определенной размерности и вида.

12. Генераторы сигналов

Последовательность импульсов низких частот (получение сигналов частот 0,5 Гц, 1 Гц, 2 Гц, ... для синтеза схем контроля и сигнализации) может быть получена с помощью генераторов периодических импульсов [2]:

$$C(k) = \bar{C}(k-1);$$

$$Y_i = \begin{cases} C(k) & \text{при } X_i = 1, i = 1, 2, \dots, KK, \\ 0 & \text{иначе,} \end{cases} \quad (22)$$

где KK — число каналов периодических импульсов, $C(k)$ — состояние выходов в момент kT , $C(k-1)$ — состояние выходов в момент $(k-1)T$, X_1, X_2, \dots, X_{kk} — состояние импульсов на входах старт/стоп генерации, Y_1, Y_2, \dots, Y_{kk} — состояние импульсов на выходах.

Заключение

Программная реализация в комплексе ПСПСЦУ основных устройств дискретной автоматики обеспечивает универсальность при проектировании цифровых САУ. Такой подход к созданию АСУ ТП позволяет без каких-либо существенных затрат расширить состав стандартизированных групп модулей дискретной автоматики по сравнению с функциональными требованиями стандарта МЭК 61131-3. Комплекс ПСПСЦУ, как средство проектирования цифровых САУ, ориентирован не на профессиональных программистов, а на инженеров-специалистов по автоматике и телемеханике. Предложенный вариант программной реализации устройств дискретной автоматики позволяет использовать методику, принятую при разработке аппаратных систем. Например, методику разработки и монтажа систем управления и регулирования на базе устройств пневмоавтоматики, гидравлики, электро-механики, аппаратуры дискретной автоматики из микросхем различной степени интеграции.

Литература

1. Konevtsov V. A., Verteshev S. M., Poletaev I. A. Eigenschaften von Complex SDSDC // European Science and Technology: 7th International scientific conference. Germany, Munich 2014, Vol. I, P. 493–497.
2. Seifart M. Digitale Schaltungen. Berlin, VEB Verlag Technik, 1986, s. 560.
3. Verteshev S. M., Konevtsov V. A., Poletaev I. A. Softwaremittel der Projektierung von Systemen der digitalen Steuerung // European Science and Technology: 5th International scientific conference. Germany, Munich 2013, Vol. I, P. 501–504.
4. Verteshev S. M., Konevtsov V. A., Poletaev I. A. Methods of Software Developing of Complex SDSDC // European Science and Technology: 4th International scientific conference. Germany, Munich 2013, Vol. I, P. 377–380.
5. Коневцов В. А. САПР цифровых САУ. Концепция: Монография. Издание третье, дополненное и исправленное: Псков: Псковский государственный университет, 2013. С. 317.
6. Лехин С. Н. Схемотехника ЭВМ. СПб.: БХВ — Петербург, 2010, 661 с.

7. Микросхемы интегральные. Серии К1500–КР1531. Справочник. С-Петербург: Издательство РНИИ «Электронстандарт», 1993, С. 130.

Об авторе(ах)

Вертешев Сергей Михайлович — доктор технических наук, профессор, заведующий кафедрой «Информационные системы и технологии», факультет информатики, Псковский государственный университет, Россия.

E-mail: president@pskgu.ru

Коневцов Владимир Александрович — кандидат технических наук, старший научный сотрудник кафедры «Информационные системы и технологии», факультет информатики, Псковский государственный университет, Россия.

E-mail: kafedravn-ist@mail.ru

Поletaев Игорь Алексеевич — старший преподаватель кафедры «Вычислительная техника», факультет информатики, Псковский государственный университет, Россия.

E-mail: ipoletaev@mail.ru

S. M. Verteshev, V. A. Konevtsov, I. A. Poletaev

DISCRETE AUTOMATIC SCHEMES FOR ASC TP

The article given shows functional possibilities of creating discrete automatic schemes in CAD of digital automatic control system (CAD of digital ACS), in complex of Software Design of System of the Digital Control. (Complex SDSDC) for automated system of control of technological processes (ASC TP). Possibilities of Complex SDSDC for implementing combination and sequential control compared with requirements of international standard IEC 61131-3:2003 (Part 3: Programming languages) are estimated.

Key words: *comparator of vectors of logical signals, flip-flop, shift register, counter of clock periods, recognition of the front of a logical signal, vote function.*

About the author(s)

Verteshev Sergey Mikhailovich, Doctor of Engineering Sciences, Professor, Head of the Department of «Information systems and technologies», Faculty of Informatics, Pskov State University, Russia.

E-mail: president@pskgu.ru

Konevtsov Vladimir Aleksandrovich, Candidate of Engineering Sciences, Senior Research Officer of the Department of «Information systems and technologies», Faculty of Informatics, Pskov State University, Russia.

E-mail: kafedravn-ist@mail.ru

Poletaev Igor Alekseevich, Senior Lecturer of the Department of «Computing machinery», Faculty of Informatics, Pskov State University, Russia.

E-mail: ipoletaev@mail.ru