

МОДУЛ ЗА ЛОГИЧЕСКО УПРАВЛЕНИЕ, РЕАЛИЗИРАН НА БАЗА CPLD

Гл. ас. инж. Стефан Билчев Белев, гл. ас. д-р инж. Йоана Емилова Русева
Русенски университет "А. Кънчев"

Logical control module, based on CPLD. The report considers implementing of an automatic mode of aggregate machine for cutting through of spindles for mixing batteries, based on CPLD. The control sequential network is described in the language for specification ABEL. The results of the simulation, as well as the part of the information about the used resources of circuit XC9536 are shown. The module gives possibility for modifying of the algorithm and provides flexibility in the control.

Въведение. Автоматичният режим на голяма част от високопроизводителните агрегатни машини в дискретното производство се характеризира с постоянен технологичен цикъл. Управляващите устройства на такъв тип машини обикновено работят като синхронни автомати с твърд цикъл на управление, реализирани на базата на програмируеми логически контролери (PLC). Универсалният характер на PLC води до наличието на определен функционален излишък (респективно и по-висока цена) на устройството за управление. На сегашния етап функциите на управляващото устройство като цяло се оптимизират посредством специализирани модули, решаващи конкретна подзадача в рамките на задачата за управление.

Едно от перспективните направления при изграждането на такъв тип модули е използването на комплексните програмируеми логически схеми (CPLD) за логическо управление на обекти и процеси през цифрови входове и изходи. Чрез тях могат да бъдат реализирани програмируеми (Firmware) логически модули, решаващи задачата за управление или нейна обособена част.

В голямата си част изградените на база CPLD управляващи устройства имат по-малко време за реакция в сравнение с традиционните PLC, което ги прави изключително удобни за изграждане на така наречените "гърбични автомати".

В доклада е разгледана реализациията на автоматичния режим на агрегатна машина за прорязване на шпиндели за смесителни батерии посредством CPLD XC9536.

Постановка на задачата. Машината има два основни елемента – манипулятор (ръка) и работна маса със захващащи устройства (стиски), посредством които детайлът се пренася последователно до отделните работни позиции, съоръжени със съответните работни органи.

Работните позиции са 4:

Позиция 1 – поемане на детайла. Работен орган – манипуляторът. Осъществява се взаимодействие между захващащото устройство на масата и манипулятора. Стиската на масата е в отворено положение. Китката на манипулятора се затваря – поема детайл от захранващия вибробункер. Ръката се

завърта на 90° , за да подаде детайла. Стиската на масата се затваря, с което детайлът е хванат. Ръката на манипулатора се отваря, т.е. освобождава се от детайла, и се връща на 0° за изчакване на нов цикъл.

Позиция 2 – прорязване на детайла. Работен орган - режеща глава 1. Извършва прорязването на детайла. Режещият инструмент е с автономно задвижване и не се управлява от разглежданото устройство. В случая елементът на управление е подаването на режещия инструмент към детайла (работен ход) и връщането му в позиция за изчакване (обратен ход). Управлението е с цикличен характер (ON/OFF), а скоростта на подаване се регулира механично. Работният ход приключва при достигане на крайно предно положение. Крайно задно положение е позицията за изчакване.

Позиция 3 – прорязване на детайла. Работен орган - режеща глава 2. Прорязва детайла по аналогичен начин, като режещият инструмент е разположен на 90° спрямо този на режеща глава 1.

Позиция 4 – разтоварване на детайла. Работен орган – механичен “изхвъргач”. Работният орган е синхронизиран механично с положението на стиската на позиция 1 и не се управлява от разглежданото устройство. “Изхвъргачът” изтласква готовия детайл от отворената стиска към изходния бункер.

Хардуерно проектиране. Задачата на хардуерното проектиране се състои в определяне на съответствието между множеството на контролираните от управляващото устройство положения на работните органи и множеството на входните сигнали на използваната CPLD, както и съответствието между множествата на подаваните от устройството управляващи въздействия и изходите на CPLD. Конкретната задача относно проектирането на интерфейса между входовете/изходите на CPLD и сензорите и изпълнителните механизми на машината не се разглежда в доклада.

Всички крайни положения на изпълнителните механизми се контролират посредством крайни изключватели и индуктивни сензори за близост, които образуват по-голямата част от входната азбука на управляващия автомат (респективно по-голямата част от множеството на входните сигнали на използваната CPLD):

- I10 и I11 – глава 1 в крайно предно, глава 1 в крайно задно;
- I12 и I13 – глава 2 в крайно предно, глава 2 в крайно задно;
- I14 и I15 – ръка отворена, ръка затворена;
- I16 и I17 – ръка на 0° , ръка на 90° ;
- I122 и I123 – подготовка за спиране ротацията на масата, позиция за спиране ротацията на масата.

В съответствие със задачата на управлението е приета положителна логика за действието на отделните сензори, т.е 1 – сензорът е активен (изпълнителният механизъм е позициониран в близост до сензора), 0 – сензорът не е активен (изпълнителният механизъм не е в близост до сензора).

Допълнителни входни сигнали към управляващото устройство се получават от пулта за управление на машината – бутони Start и Stop.

Синхронизацията на устройството се извършва посредством тактовия сигнал C1, получен от външен тактов генератор.

Изпълнителните механизми на всички работни органи са с циклично управление (ON/OFF), което е предпоставка за използване изходите на включените в състава на използваната схема тригери Q_i в качеството на източници на управляващи въздействия:

Q1 – отваряне (1)/затваряне на клещите (0);

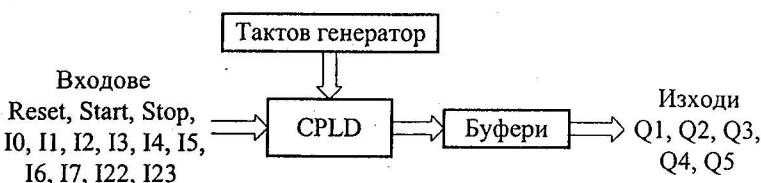
Q2 – движение на режещите глави напред (1)/назад (0);

Q3 – отваряне (1) /затваряне на ръката (0);

Q4 – въртене на ръката от 0⁰ към 90⁰ (1)/ въртене на ръката от 90⁰ към 0⁰ (0);

Q5 – въртене на масата (1)/маса в установлено положение (0).

Блоковата схема на управляващото устройство е показана на фиг. 1.



Фиг. 1. Блокова схема на модула за логическо управление

Програмиране на CPLD. Съществен елемент при използването на CPLD е системата за тяхното програмиране, тъй като описането на задачата за управление на езика на конкретната схема в известна степен води до качествено ново ниво на програмиране.

В общия случай задачата на програмирането е завършващ етап от синтеза на управляващото устройство. За целта функционирането на управляващия автомат трябва да се “преведе” от езика за проектиране, в случая функционална блокова схема (фиг. 2), на разбираем за използваната CPLD език. Проектирането се извършва при определени ограничения, свързани с наличния ресурс на използваната схема, предоставените възможности за кодиране на състоянията на управляващия автомат и възможностите за синтезиране на възбудителните функции на отделните елементи памет.

В случая е използвана развойната система Xilinx Foundation Series [1, 3]. Тя предоставя два графични и един текстов редактор за въвеждане на проекта. Използваната CPLD (XC9536-15), съдържа 36 макроклетки, организирани в 2 функционални блока [2]. Автоматът е описан на езика за спецификация ABEL чрез текстовия редактор HDL Editor (фиг. 3.).

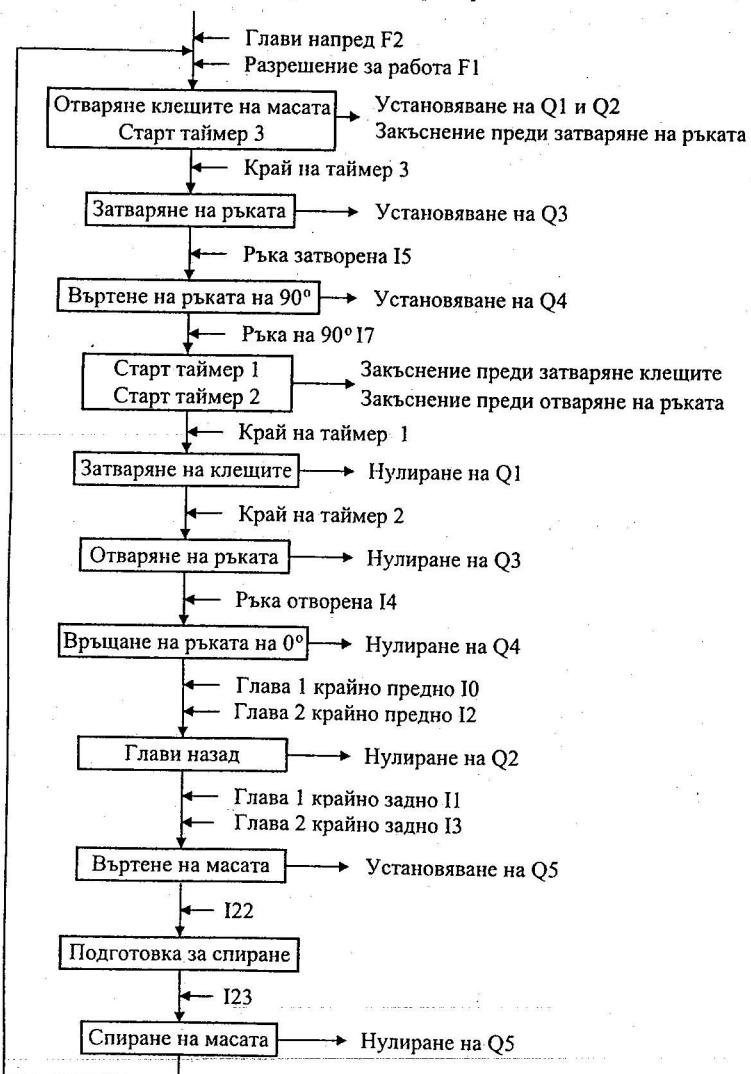
Ред 4 описва входовете на автомата.

Въведени са вътрешни тригери F_i, които дават съответни условия:

- F0 – стартиране на нов цикъл;

- F1 – разрешение за работа;

- F2 – за движение на режещите глави напред;
 - F3 – крайно предно положение на главите и ръка, завъртяна на 90^0 .
- Множеството вътрешни променливи включва вътрешните тригери $F3 + F0$ и тригерите с изходи, изведени на изводите на схемата – $Q5 + Q1$. Тези тригери са програмирани като синхронни S-R тригери.



Фиг. 2. Функционална блокова схема

```

2. Title 'CPLD'
3. Declarations
4. Reset, Start, Stop, Cl, I0, I1, I2, I3, I4, I5, I6, I7, I22, I23 PIN;
5. Q5..Q1 PIN istype 'reg_sr'; F3..F0 node istype 'reg_sr';
6. T5..T0, C5..C0 node istype 'reg'; CEC , CET node istype 'reg_sr';
7. Q=[Q5..Q1]; F=[F3..F0]; T=[T5..T0]; C=[C5..C0];
8. SF3..SF0,RF3..RF0,SQ5..SQ1,RQ5..RQ1 node istype 'com';
9. SF=[SF3..SF0]; RF=[RF3..RF0];
10. SQ=[SQ5..SQ1]; RQ=[RQ5..RQ1];
11. Equations
12. [C,T,F,Q,CEC,CET].CLK=Cl; [C,T,F,Q].ar=Reset;
13. CET.s=F1&F2&I7; CEC.s=F1&F2; CET.r=RQ3; CEC.r!=F2;
14. when CEC then C:= C+1; else C := 0;
15. when CET then T:= T+1; else T := 0;
16. [F].s=[SF]; [F].r=[RF]; [Q].s=[SQ]; [Q].r=[RQ];
17. SF3=I0&I2&I6; RF3=Stop#I22;
18. SF2=F1&(Start#I23); RF2=I0&I2;
19. SF1=Start&I23&I1&I3&I6&!Q1; RF1=Stop;
20. SF0=I22; RF0=I23;
21. SQ1=F1&F2; RQ1=T5&!T4&T3&!T2&!T1&!T0;
22. SQ2=F1&F2; RQ2=F1&I0&I2;
23. SQ3=C5&!C4&C3&!C2&!C1&!C0;
RQ3=T5&T4&!T3&!T2&!T1&!T0;
24. SQ4=F1&F2&I5; RQ4=F1&F3&I4;
25. SQ5=F1&F3&I1&I3&I6; RQ5=F0&I23;
26. end CPLD

```

Фиг. 3. Описание на автомата на езика ABEL

Редове 17÷25 задават логическите условия на входове SF_i , SQ_i (за поставяне на съответния тригър в единично състояние) и RF_i , RQ_i (за нулиране на съответния тригър). За реализиране на трите таймера са използвани два 6-разрядни синхронни брояча – Т и С, описани с редове 14 и 15. Брояч Т се използва за таймери 1 и 2, които се стартират едновременно, но изработват различни времезадължения. (Разрядността на броячите се определя от тактовата честота.)

Част от командният файл за симулиране работата на автомата е показан по-долу, а резултатите от симулацията – на фиг. 4.

delete_signals

restart

vector C C5_Q C4_Q C3_Q C2_Q C1_Q C0_Q

vector T T5_Q T4_Q T3_Q T2_Q T1_Q T0_Q

watch Cl Reset Start I0 I1 I2 I3 I4 I5 I6 I7 I22 I23

watch F0_Q F1_Q F2_Q F3_Q Q1 Q2 Q3 Q4 Q5

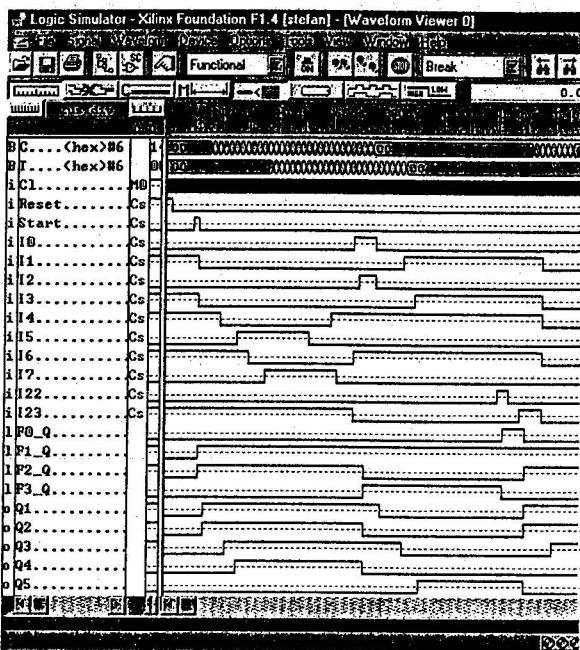
wfm I0 @0us=0 @68us=1 @75us=0 @135us=0

wfm I1 @0us=1 @12us=0 @85us=1 @135us=0

wfm I2 @0us=0 @70us=1 @75us=0 @135us=0

clock Cl 0 1

stepsize 500ns



Фиг. 4. Резултати от симулацията

Част от получената чрез Fitter Report [3] информация за използваните ресурси на схема XC9536 е показана по-долу:

*****Resource Summary*****				
Device	Macrocells	Product Terms	Pins	
XC9536-15PC44	23 /36 (64%)	41 /180 (23%)	19 /34 (55%)	

Заключение. Проведената функционална симулация доказва коректното функциониране на автомата. Конфигурацията на устройството за управление на прорязването на детайла е минимална. Използваната CPLD притежава достатъчен ресурс за модификация на управлението, включително и за добавяне на управление на ръчния режим на работа.

Литература:

1. Гиздарски Е., Проектиране с програмируема логика, Русе, 1998.
2. The programmable logic, Data Book, Xilinx, 1998.
3. Xilinx Foundation Series.