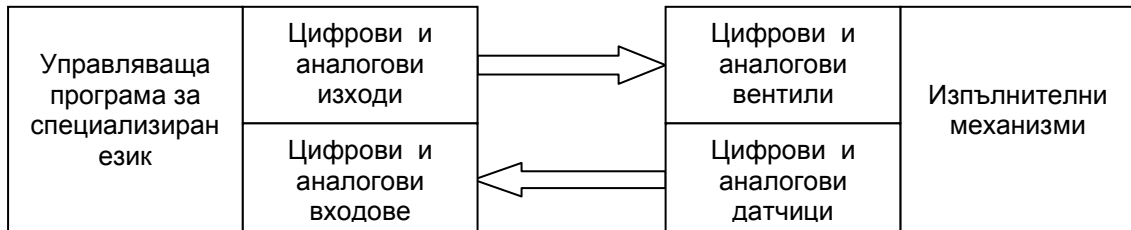


УПРАЖНЕНИЕ № 1

ПРОГРАМИРУЕМИ ЛОГИЧЕКИ КОНТРОЛЕРИ ЗА УПРАВЛЕНИЕ НА ИНДУСТРИАЛНИ ПРОЦЕСИ ЕЗИК ЗА ПРОГРАМИРАНЕ LADDER

Обобщена схема за управление на промишлен процес чрез програмируем логически контролер е показана на фигура 1.

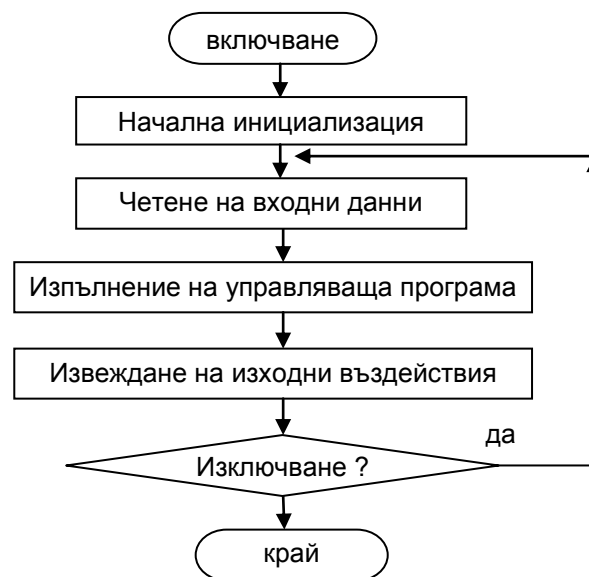


Фиг. 1: Управление на индустриален процес чрез програмируем логически контролер

Управляващото устройство /УУ/ е индустриален компютър със специфична периферия. Управляващият алгоритъм е заложен в програма, написана на специализиран език. Изходните въздействия към управлявания обект се изпращат чрез цифрови и аналогови изходи. Данни за текущото му състояние постъпват чрез цифрови и аналогови входове.

Управляваният обект /УО/ е съставен от набор изпълнителни механизми. Те се задействат чрез аналогови и цифрови активиращи вентили - релета, хидравлични или пневматични клапани, електродвигатели и др. Обектът генерира данни за своето текущо състояние чрез набор от цифрови и аналогови датчици.

Връзката между УУ и УО се осъществява чрез два интерфейса. Първият интерфейс подава изходни въздействия от УУ към УО, а вторият – данни за състоянието на обекта от УО към УУ.



Фиг.2: Обобщен алгоритъм на работа на програмируем логически контролер

Обобщен алгоритъм на работа на програмируем контролер е показан на фигура 2.

Стандартът PL7/2 допуска използването на два специализирани езика за създаване и въвеждане на управляващата програма:

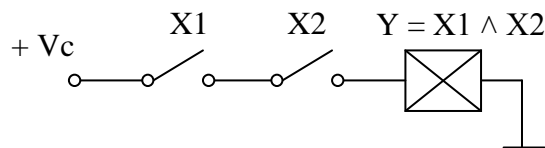
- LADDER – графичен език за въвеждане на релейно- контакторни схеми, отразяващи управляващия алгоритъм. Съдържа контактори, бобини на релета и специализирани функционални блокове;
- GRAFCET – графичен език за въвеждане на крайни автомати, отразяващи управляващия алгоритъм. Съдържа крайни състояния наречени етапи и преходи с логически условия.

Въведение в релейно контакторната логика

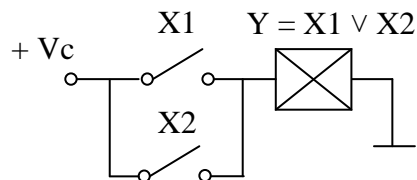
Известно е, че произволна логическа функция може да бъде представена в нормална дизюнктивна форма, чрез трите базови логически функции – конюнкция, дизюнкция и инверсия.

Тези логически функции, от своя страна, могат да бъдат реализирани апаратно чрез контакти и бобини на релета, съгласно схемите представени на фигури 3,4 и 5.

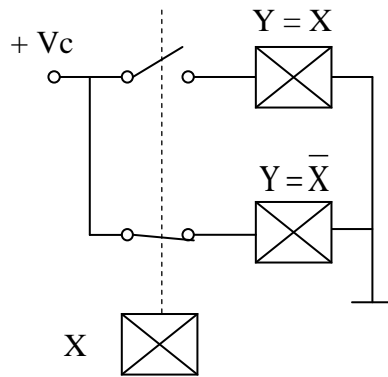
Следователно, всяка една логическа функция може да бъде реализирана чрез подобна релейно- контакторна логика.



Фиг. 3: Изпълнение на логическа функция дизюнкция чрез релейно контакторна схема



Фиг. 4: Изпълнение на логическа функция конюнкция чрез релейно контакторна схема



Фиг. 5: Изпълнение на логическа функция инверсия чрез релейно контакторна схема

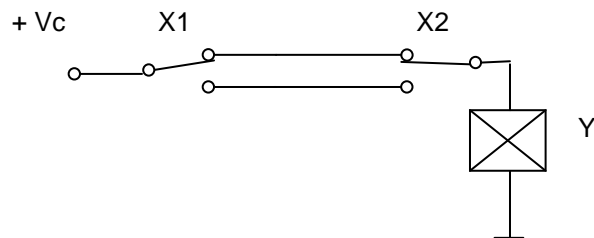
Графичен език LADDER

Езикът LADDER се базира на подхода за въвеждане на управляващия алгоритъм под формата на множество паралелно изпълняващи се логически функции. Тези функции се описват графично чрез релейно- контакторни схеми.

Типовете и наименованията на адресируемите променливи, както и съответните им графични елементи са описани в приложение 1, стр. 1÷5 и стр. 11.

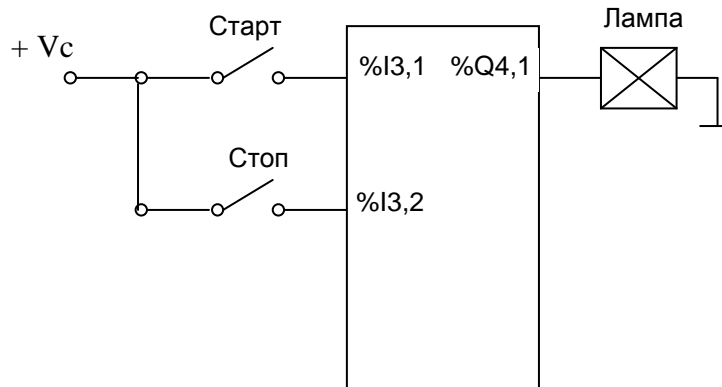
Задачи

1. Да се реализира управление на девиаторен електрически ключ, съгласно принципна електрическа схема от фигура 6:

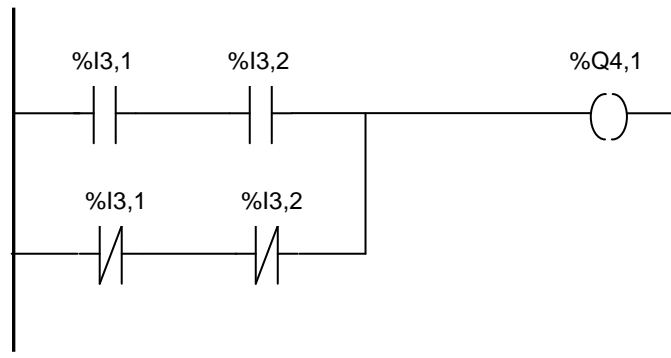


Фиг. 6: Принципна електрическа схема на девиаторен ключ

Решение:

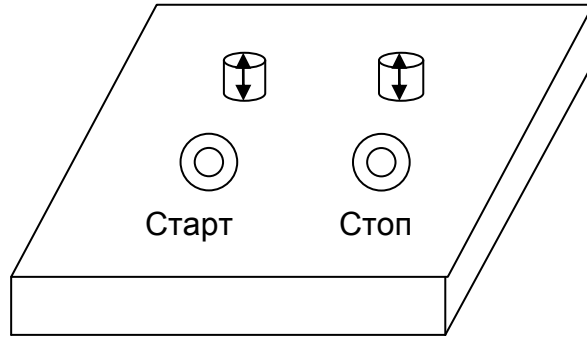


Фиг. 7: Схема на свързване на електрическите ключове и лампата към програмируемия контролер



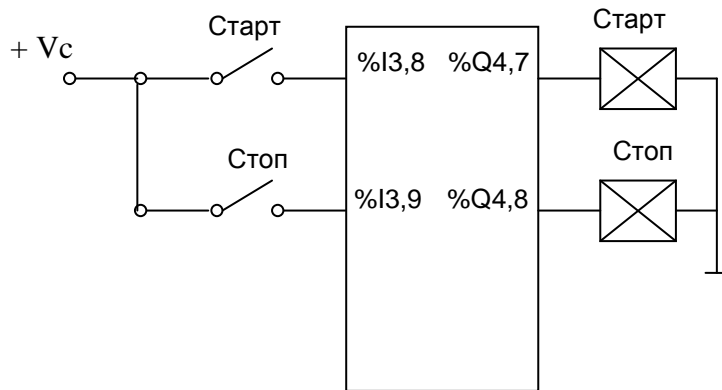
Фиг. 8: Управляващ алгоритъм на език LADDER

2. Да се реализира прост пулт за управление на промишлен процес съгласно фигура 7, със следните характеристики:
 - пултът се състои от 2 бутона и 2 светлинни индикатора с наименования „Старт“ и „Стоп“;
 - бутоните служат за избор на състояние, а индикаторите показват текущо избраното състояние;
 - изборът на състояние се извършва чрез натискане на съответния бутон;
 - избраното състояние остава активно и след отпускане на бутона за избор;
 - началното състояние след включване е „Стоп“;
 - при едновременни натискане на двата бутона предимство има състояние „Стоп“

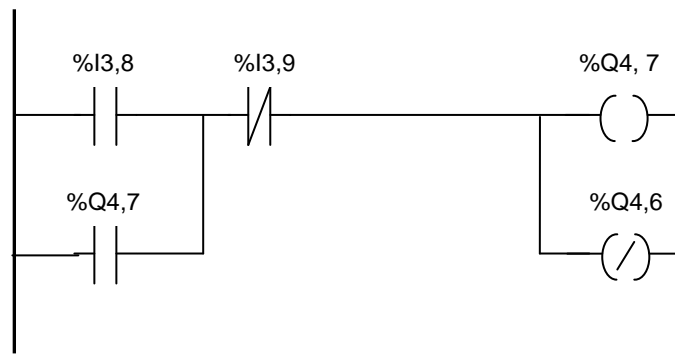


Фиг. 9: Външен вид на прост пулт за управление на промишлен процес

Решение:



Фиг. 10: Схема на свързване на датчици и изходни устройства към програмируемия контролер



Фиг. 11: Управляващ алгоритъм на език LADDER