



Софийски университет "Св. Климент Охридски"
Факултет по математика и информатика

РАЗПРЕДЕЛЕНО УПРАВЛЕНИЕ НА КОНВЕЙЕРНИ СИСТЕМИ

ДИСЕРТАЦИЯ

за присъждане на образователна и научна степен
„ДОКТОР“

по професионално направление 4.6 Информатика и
компютърни науки

Докторска програма „Компютърни науки“

Докторант: Маг. инж. Ивайло Тодоров Андонов
Научен ръководител: Доц. д-р инж. Симеон Емилов Цветанов

София 2021

Благодарности

Искам да изкажа моите най-големи благодарности на всички, които ме подкрепяха и допринесоха за създаването и довършването на тази работа.

Моят ръководител доц. Симеон Цветанов с много голям принос за оформяне на академичния вид на работата и за да придобие тя вид на дисертация, както и за инициативите му за направените публикации.

Компанията Индустиален Софтуер АД и нейният управител Димитър Петров, за възможността и предоставените технически средства за извършване на експерименти като модули, конвейери и апаратура.

Проф. Иван Петров, за неговите насоки и напътствия с научен поглед и за предоставените материали за моторизирани ролери.

Административния състав на факултета по математика и информатика на Софийски Университет Климент Охридски, който често ме подсещаше важни срокове и беше благосклонно настроени и с градивна критика ме насърчаваше да продължавам.

И не на последно място моят брат и моето семейство, които ми бяха упора в много моменти и ме мотивираха да продължавам.

Съдържание

Списък с фигури	6
УВОД	9
I ГЛАВА. ОСОБЕНОСТИ ПРИ УПРАВЛЕНИЕТО НА КОНВЕЙЕРНИ СИСТЕМИ	10
I.1 Видове конвейерни системи според начина на придвижване на обектите 10	
I.2 Видове конвейерни системи според задвижването.....	11
I.3 Видове конвейери според управлението на зоните	12
I.4 Видове конвейери според алгоритъма на управление на зоните 14	
I.5 Изисквания към управлението на конвейерни системи	16
I.6 Разпределени системи	17
I.7 Проблеми при захранването на конвейерни системи с МР....	19
I.7.1 Инсталирана мощност	19
I.7.2 Върната енергия	20
I.8 Проблеми с надеждността на данните в силно зашумена среда 22	
I.9 Проблеми свързани със защитата на интелектуалната собственост вложена в модулите за управление	22
I.10 Софтуерни агенти	25
I.11 Изкуствен интелект и машинно обучение	26
I.11.1 Невронни мрежи.....	29
I.12 Компютърни симулации	29
I.13 Заключение	31
I.14 ЦЕЛ И ЗАДАЧИ НА ДИСЕРТАЦИОННИЯ ТРУД.....	32
II ГЛАВА. РЕШЕНИЯ ЗА ДОБАВЯНЕ НА ИНТЕЛИГЕНТНОСТ КЪМ СИСТЕМИТЕ ЗА УПРАВЛЕНИЕ НА КОНВЕЙЕРИ.....	33
II.1 Предложение за подобряване ефективността на конвейерни системи, чрез прилагане на алгоритъм разпределено управление	33
II.1.1 Архитектура на предложената разпределената система за управление 33	
II.1.2 Предложение за алгоритъм за разпределено управление.	36
II.1.3 Разпределено изчисляване на целевата функция	37
II.1.4 Реакция при неочаквани събития.....	38

II.1.5	Допълнителен модул за разпознаване на изображение и измерване	40
II.2	Предложение за разпределяне на електроенергията между зоните	41
II.3	Методи за подобряване на комуникацията между мотора и контролера	49
II.4	Защита на устройствата от клониране	50
II.4.1	Предизвикателства при копирането на модулите за управление	50
II.5	Добавяне на интелигентност чрез използване на невронна мрежа	55
III	ГЛАВА. ИМПЛЕМЕНТАЦИЯ НА ПРЕДЛОЖЕНИТЕ РЕШЕНИЯ	58
III.1	Имплементация на предложеният метод за подобряване качеството на данните за позицията на моторите	58
III.2	Имплементация предложения метод за подобряване на захранването	62
III.3	Имплементация на метода за защита на модулите от клониране	65
III.4	Имплементация на предложения метод за разпределено управление чрез компютърни симулации	67
III.5	Имплементация на предложения метод за разпределено управление върху реална система	69
III.5.1	Актуатори за конвейерни системи	69
III.5.2	Безчеткови постояннотокови мотори използвани за задвижване на конвейери	71
III.5.3	Безсензорни конвейери	79
III.5.4	Приложение на предложения метод	81
III.6	Заключение	Error! Bookmark not defined.
IV	ГЛАВА. ЕКСПЕРИМЕНТИ И АНАЛИЗ НА РЕЗУЛТАТИТЕ	84
IV.1	Експерименти с предложият метод за подобряване качеството на данните за позицията на моторите	84
IV.2	Експерименти с предложия метод за подобряване на захранването	85
IV.3	Експерименти с предложия метод за разпределено управление	87

IV.3.1	Очаквани резултати.....	87
IV.3.2	Резултати от симулациите.....	87
IV.4	Експерименти и тестове на цялата система.....	90
IV.5	Заключение.....	92
Приноси	93
Публикации	94
ИЗПОЛЗВАНА ЛИТЕРАТУРА	95
Декларация за оригиналност на резултатите	100

Списък с фигури

Фиг. 1 Централизирано задвижване на еднозонов конвейер	10
Фиг. 2 Централизирано задвижване на тризонов конвейер	11
Фиг. 3 Разпределено задвижване	12
Фиг. 4 Централизирано управление с PLC	13
Фиг. 5 Разпределено управление	13
Фиг. 6 Позициониране на без гарантирано разстояние между обектите	15
Фиг. 7 Гарантирано разстояние между обектите	15
Фиг. 8 Разположение на обектите спрямо зоните	17
Фиг. 9 CAP теорема	18
Фиг. 10 Захранване на конвейера.....	19
Фиг. 11 Примерна консумирана мощност в течение на времето	20
Фиг. 12 Примерно сработване на защита от пренапрежение	21
Фиг. 13 Датчици на Хол монтирани на печатна платка	22
Фиг. 14 Клонирано и оригинално устройство	23
Фиг. 15 Инвестиции на производителя на контролера.....	24
Фиг. 16 Сравнение на инвестициите за разработване и обратно инженерство.....	25
Фиг. 17 Структура на единиците на LSTM невронна мрежа	29
Фиг. 18 Създаване на модел в FlexSim.....	30
Фиг. 19 Симулиране на конвейерна система с кръстовище.....	30
Фиг. 20 Симулиране на сортировъчен център.....	31
Фиг. 21 Измерване на дължината на зоната	34
Фиг. 22 Всяка зона запазва три списъка с обекти	35
Фиг. 23 Скъсяване на празното разстояние по време на приемане на обект	37
Фиг. 24 Неочаквано събитие	39
Фиг. 25 Допълнителен модул за измерване на размери използващ камера	40
Фиг. 26 Времедиаграма на получените съобщения	41
Фиг. 27 Примерна схема за захранване и комуникация на конвейер .	42
Фиг. 28 Прост пример за разпределяне на захранването	45
Фиг. 29 Реалистичен пример за разпределяне на захранването	46
Фиг. 30 Видове софтуер в постоянната памет на контролера	51
Фиг. 31 Архитектура на контролер управляващ четири зони	52
Фиг. 32 Използване на половината вътрешна памет за временно място по време на обновяване на софтуера	53
Фиг. 33 Използване на външна памет за временно място при обновяване на софтуера.....	53
Фиг. 34 Само-обновяване на софтуера от съседните модули.....	54
Фиг. 37 Структура на Feed-forward невронна мрежа.....	56

Фиг. 38 Ефект от игнорирането на погрешна информация от датчиците за позиция	58
Фиг. 39 Ефект от приемането на следващата по посока на въртене информация при получаване на грешна такава.	59
Фиг. 40 Влияние на получената грешна информация за положението на ротора върху консумирания ток от мотора	60
Фиг. 41 Имплементация на софтуера генериращ комбинациите по Хаминг.....	61
Фиг. 42 Резултат от работа на програмата намиращата равно отдалечени комбинации по Хаминг	62
Фиг. 43 Формат на съобщението за консумиран ток.....	63
Фиг. 44 Повишаване на захранващото напрежение при спиране на малък обект.	64
Фиг. 45 Повишаване на захранващото напрежение при спиране на тежък обект.	65
Фиг. 35 Криптиране на данните	66
Фиг. 36 Изграждане на пръстов отпечатък на устройството, което се използва за криптиране на фърмуера и данните.....	67
Фиг. 46 Код за реализиране на динамично променяне на празното разстояние между обектите	68
Фиг. 47 Задаване на параметри на вградената функционалност за създаване на празно разстояние между обектите	69
Фиг. 48 Напречно избутване	70
Фиг. 49 Изскачащи колелца.....	70
Фиг. 50 Завъртаща се маса.....	71
Фиг. 51 Устройство за напречен трансфер с повдигащи се ролки	71
Фиг. 52 Устройство за напречен трансфер с повдигащи се ремъци ...	71
Фиг. 53 Типични моторизирани ролери	72
Фиг. 54 Блокова схема на МР	72
Фиг. 55 Основни компоненти на моторизиран ролер.....	73
Фиг. 56 Компоненти на мотор-скоростна кутия	74
Фиг. 57 Разрез на БЧПТ двигател (С благодарност към Киьова Европа ЕООД)	74
Фиг. 58 Опростена диаграма на семейството електрически машини ..	75
Фиг. 59 Статор и ротор на БЧПТ мотор	76
Фиг. 60 Илюстрация на принципа на работа на БЧПТ мотор	77
Фиг. 61 Конвейер с фотосензори	79
Фиг. 62 Безсензорен конвейер.....	80
Фиг. 63 Безсензорен конвейер с безсензорни БЧПТ мотори.	81
Фиг. 64 Конвейер с малка плътност на обектите	81
Фиг. 65 Конвейер с голяма плътност на обектите	81
Фиг. 66 Конвейер с перпендикулярно придвижване	82

Фиг. 67 Осцилограма на напрежението при спиране с използване а новия метод за разпределение на енергията	86
Фиг. 68 Ниска плътност на обектите	87
Фиг. 69 Висока плътност на обектите	87
Фиг. 70 Конфигуриране на симулацията	88
Фиг. 71 Симулиране на конвейер със различни по размер обекти	88
Фиг. 72 Работа на конвейера без акумулиране на обектите.....	89
Фиг. 73 Акумулация на конвейера след преминаването на 20 обекта	89
Фиг. 74 Експериментален конвейер	90
Фиг. 75 Входяща част от конвейер в център за сортиране на пратки.	91
Фиг. 76 Рециркулираща част от конвейер	91

УВОД

Основна цел на дисертацията е разработването на решение за подобряване ефективността на конвейерни системи [1], чрез прилагане на нови методи за разпределено управление.

Предложеното решение разглежда отделните зони на конвейера като софтуерни агенти, които чрез собственото си поведение се стремят към подобряване на като средната скорост и гъстотата придвижваните обекти на цялата конвейерна система.

Предложен е нов алгоритъм, който се опитва да постигне висока средна скорост и гъстота придвижваните обекти използвайки информация за позицията на двигателите, изчислителната способност на модулите за управление на зоните, камери с достъпна цена и високата скорост на комуникационната мрежа.

Целта е да се извлече възможната полезна информация и да се използва по най-добрия начин, за да се подобри ефективността на конвейерната система.

Приносите на дисертацията са от полза за компаниите занимаващи се с инженерингова дейност по конструиране на системи включващи конвейери за автоматизираните складове, центровете за сортиране, летища, магазини, различни производствени предприятия и др., като им позволява да намалят общата им себестойност и да ги направи по-интелигентни.

Дисертационния труд е оформен в четири глави.

Първа глава прави анализ на предметната област и завършва с дефиниране на целите и задачите на дисертационния труд.

Във **Втора глава** се предлага решение за разпределено управление, където зоните се разглеждат като софтуерни агенти, които чрез собственото си поведение се стремят към подобряване на глобалните характеристиките на цялата система. Предложени са също и решения за подобряване на подсистемите за захранване, измерване и комуникация.

Трета глава представя имплементацията на отделните компоненти на системата и тяхната интеграцията.

Четвърта глава е посветена на тестване функционирането на системата посредством експериментални изследвания и е разделена на две части, като в първата са представени тестовете на отделните компоненти на платформата, а във втората резултатите от тестовете верифициращи интеграцията на цялата система.

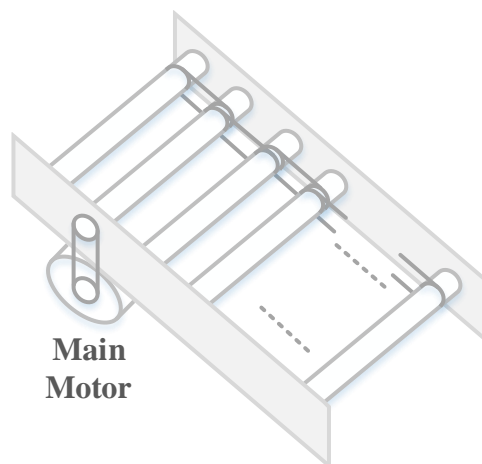
I ГЛАВА. ОСОБЕНОСТИ ПРИ УПРАВЛЕНИЕТО НА КОНВЕЙЕРНИ СИСТЕМИ

Съвременните конвейерни системи намират все по-важна роля при придвижването на товари в автоматизираните складове, центровете за сортиране, летища, магазини, различни производствени предприятия и др.

Задачите, които конвейерните системи изпълняват нараства, което води до увеличаване на тяхната сложност, същевременно цената на конвейерите става все по-ниска, а пазарът на такива системи все по-голям.

I.1 Видове конвейерни системи според начина на придвижване на обектите

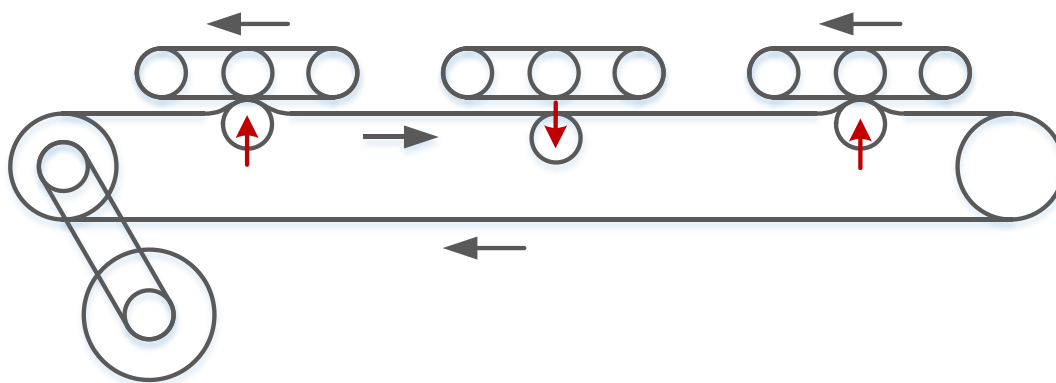
Най-простият вид конвейери представлява група ролки или лента задвижвани едновременно обикновено от електродвигател [2] Фиг. 1. Това представлява една зона за придвижване. Начинът по който обектите са поставени в началото, като ориентация и разстояние по между им остава непроменен до края на придвижването им.



Фиг. 1 Централизирано задвижване на еднозонов конвейер

Обикновено при тези конвейери има един фотосензор на края на им за отчитане наличието на пристигнали обекти. При липса на готовност за приемане на обектите от отсрещната страна, конвейера спира движението си и изпада в т.н. режим на акумулация.

При другият вариант, в който конвейера е образуван от много зони Фиг. 2 и Фиг. 3 разстоянието между обектите **може** да се променя по време на придвижването им. В този случай обикновено има фотосензор в края на всяка зона, за отчитане пристигането или напускането на обект. В настоящата работа има интерес да се разглежда втория вид, защото предлага възможности за реализация на различни алгоритми променящи ориентацията и взаимното разположение на обектите.



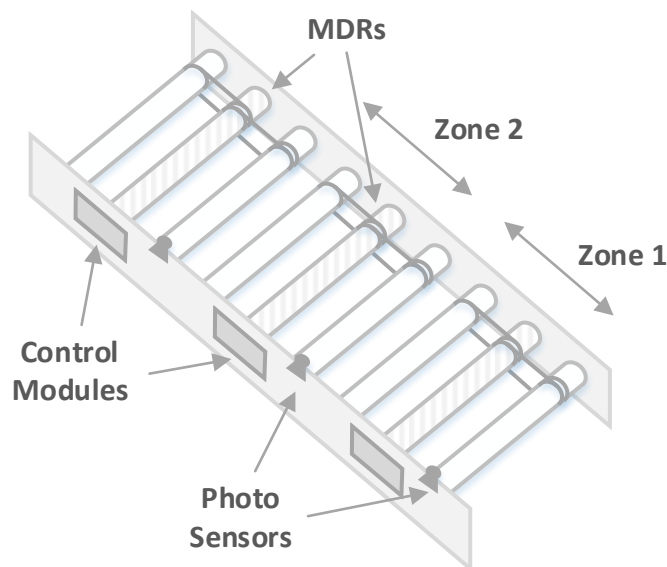
Фиг. 2 Централизирано задвижване на тризонов конвейер

1.2 Видове конвейерни системи според задвижването

В зависимост от начина на задвижване конвейерите могат да бъдат разделени на такива с централизирано задвижване Фиг. 1, Фиг. 2 от един електродвигател, като обикновено той задвижва дълъг ремък, към който може да бъде зацепвана (обикновено една) ролка от зоната с пневматичен или електромагнитен актуатор [3], [4]. Останалите ролки от зоната са свързани със зацепената посредством ремъци. Вторият вид задвижване е разпределено от много електродвигатели обикновено вградени в моторизирани задвижващи ролки – (MDRs) Фиг. 3, [5].

Две са основните причини, заради които конвейерите със зони [6] за предпочитани в съоръженията, при които има наличие на работещи хора. [7]. Първата е безопасност. Когато конвейера е изграден с централизирано задвижване е необходима мощност, която да е достатъчна за задвижването на цялата линия. Обикновено тя е многократно по-голяма от мощността на моторизирания ролер при конвейерите с разпределено задвижване [8], [9]. Поражения върху човек, като прищипване на ръцете от придвижваните обекти или навиване на коса или дрехи по въртящите се ролки са много по-тежки при конвейерите с централизирано задвижване в сравнение с тези с разпределено задвижване

Друг аспект на безопасността е ограничената големина на захранващото напрежение до 60V стигащо до електроапаратура намираща се извън метални шкафове или в частност електродвигателя. При използването на мощен електродвигател за централизирано задвижване големината на тока е значителна и сечението на кабелите е голямо, което допълнително води до оскъпяване на конвейера.



Фиг. 3 Разпределено задвижване

Втората причина е ефективността [10]. В първият случай се налага задвижването на цялата линия, дори при придвижването на само един обект от началото до края. Във втория случай (на разпределено задвижване) се задвижват само ролките от зоната върху която е обекта, което изисква многократно по-малка мощност.

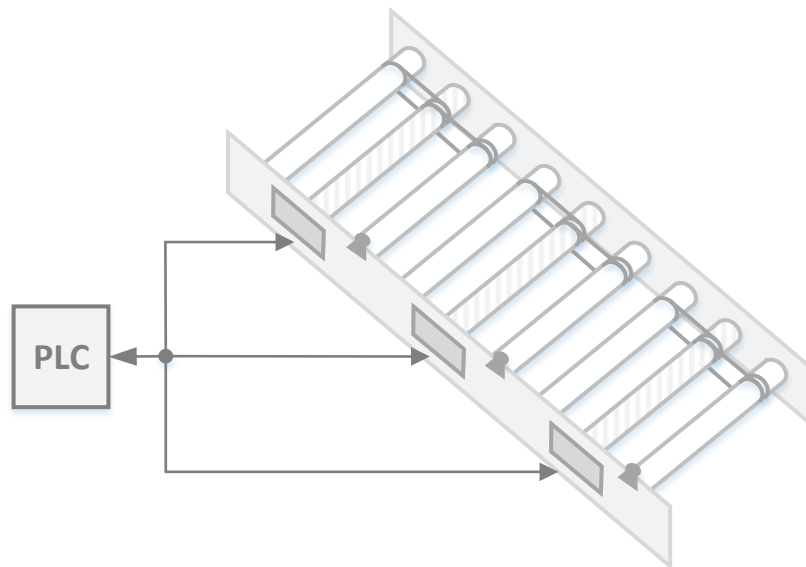
При централизираното задвижване на конвейери с повече от една зона, задвижването на конкретна зона става рязко с недетерминирано ускорение, защото актуатора / съединителя зацепва ролките на зоната към вече въртящ се с определена скорост ремък. Това може да предизвика приплъзване на обекта по конвейера, което внася неопределеност в резултатите от алгоритмите. Освен това, както беше вече обяснено тези конвейери не са предпочитани по други причини. Именно затова конвейерите с разпределено задвижване представляват интерес за настоящото проучване.

1.3 Видове конвейери според управлението на зоните

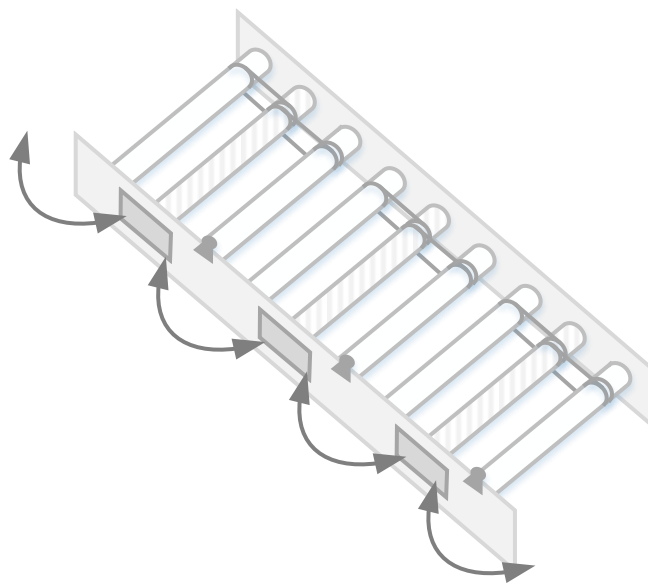
Според начина на управление конвейерите с разпределено задвижване (наричани на кратко конвейери от тук нататък) може да бъде разделен на два вида. Централизирано, където логиката за управление на всички зони е имплементирана в едно устройство Фиг. 4.

Другият вид управление е децентрализирания, при който контролера за управление на моторизирания ролер имплементира и логиката за управление на зоната Фиг. 5. Обикновено като устройство за централизирано управление се използва Програмируем Логически Контролер (от англ. PLC), което освен да изпълнява логиката за управление на зоните извършва и друго управление свързано с целия производствен

процес в който конвейера участва. Въпреки това обаче, при наличие на конвейерна част с много зони значителна част от изчислителните ресурси на контролер биват изчерпани само за конвейерната част, което налага използването на по-мощно PLC, което е и по-скъпо.



Фиг. 4 Централизирано управление с PLC



Фиг. 5 Разпределено управление

Развитието на електрониката позволява използването на все по-мощни контролери (с по-голяма постоянна и оперативна памет и по-голяма скорост на изпълнение, дори двуядрени и с отделни комуникационни копроцесори)

за една и съща задача, което означава, че в контролера управляващ моторизирания ролер има достатъчно ресурси за изпълняване на още задачи – напр. логика за управление на зоните.

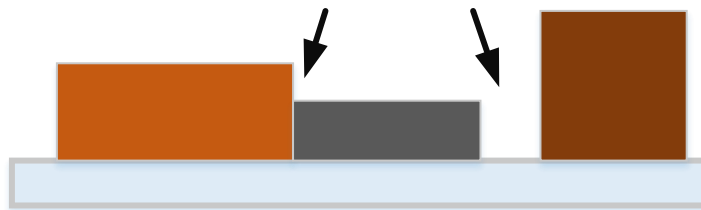
Преместването на част или цялата логика на управлението на зоните в контролерите на по-ниско ниво представлява интерес за индустриалните среди и настоящото проучване по няколко причини:

- Така отпада изцяло или почти изцяло имплементирането на логика за управление на зоните в програмируемите контролери, което обикновено се прави за всяка система индивидуално и изисква времеви и финансов ресурс от компанията интегрираща системата.
- Необходимостта от по-мощно PLC отпада и така системата поевтинява.
- Разработените в течение на времето нови алгоритми за управление, както предложените в настоящата работа са вече заложили в новите контролери за моторизирани ролери и системния интегратор получава функционалност без допълнителни разходи.
- При свързване на две различни системи с конвейер отпада необходимостта от програмируем контролер.

1.4 Видове конвейери според алгоритъма на управление на зоните

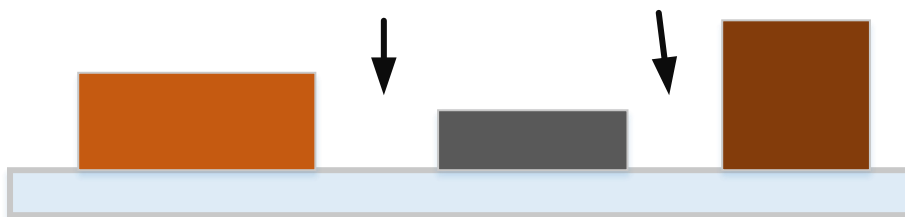
Конвейерите с централизирано задвижване представляващи една зона не предлагат възможност за различни режими на придвижване. В конвейерите образувани от зони съществуват няколко режима на придвижване на обектите, които са се наложили, заради различни техни характеристики:

- Едновременно придвижване без гарантирано разстояние (от англ. slug mode) Фиг. 6, при който начина, по който обектите са разположени в началото остава непроменен до края. Единствената разлика спрямо конвейерите с една зона е, че зоните върху които няма обекти не се задвижват, което осигурява някаква енергийна ефективност. Недостатък е, че като се появи необходимост обектите да бъдат придвижени по-нататък, всички зони върху които има обекти и директно следващите ги, които трябва да поемат обектите биват задвижени едновременно. Това предизвиква голяма внезапна консумация на ток и затова захранващите блокове трябва да са оразмерени правилно по този показател.



Фиг. 6 Позициониране на без гарантирано разстояние между обектите

- Едновременно придвижване с гарантирано разстояние (от англ. Zero Pressure Accumulation Train mode или ZPA train) Фиг. 7, който се отличава от предишния режим по това, че когато фотосензор отчете напускане на обект и последващо пристигане на такъв тя може да спре за момент, за да увеличи разстоянието между двата обекта. Това създава минимално разстояние между обектите, което е параметър на конвейера, който може да се конфигурира.



Фиг. 7 Гарантирано разстояние между обектите

- Последователно придвижване с гарантирано разстояние (от англ. ZPA singulation mode) Фиг. 7, при който всяка зона подава обект към следващата зона, следи за пристигането му върху нея (следи състоянието на следващата зона) и не подава повече обекти, докато обекта не пристигне и напусне зоната. При напълнен конвейер с обекти, придвижването на всички с една позиция напред изглежда, като придвижването на едно свободно място назад. Предимство на този режим е, че вероятността всички зони да трябва да бъдат задвижени по едно и също време е много малка.

- Смесен режим, при който зоните могат да уплътняват или отдалечават взаимното разстояние между обектите в зависимост от различни критерии, които са счетени за важни в конкретния момент. Този режим представлява интерес за настоящата работа, защото предлага възможност за реализация на различни алгоритми за оптимизиране на работата на цялата система.

1.5 Изисквания към управлението на конвейерни системи

Освен основната цел да придвижи някакво количество обекти от една до друга точка в пространството от конвейерната система се очакват още функции, като буфериране на обектите, промяна на ориентацията (вертикалното завъртане), дори пренареждане на подредбата на обектите. Някои такива функции позволяват автономно изпълнение от конвейера, като той ползва информацията за пристигащи обекти от входа, готовността за приемане на такива на изхода му, както и наличната сензорна информация от всички зони, изграждащи участъка.

В реалните системи себестойността на конвейера е винаги важна, затова при участъци с праволинейно движение и когато физическите характеристики на обектите позволяват се прибегва до използването на безсензорни зони [11]. В този случай информация за пристигнал и напуснал обект се извлича от наблюдател (оценител) на динамиката на задвижвания от мотора товар, съставен от ролките на зоната и придвижвания обект. Контролерът на зоната прави идентификация на товара, при начално пускане на системата и периодично в бъдеще, за компенсирание на измененията в тези параметри от износване и други фактори.

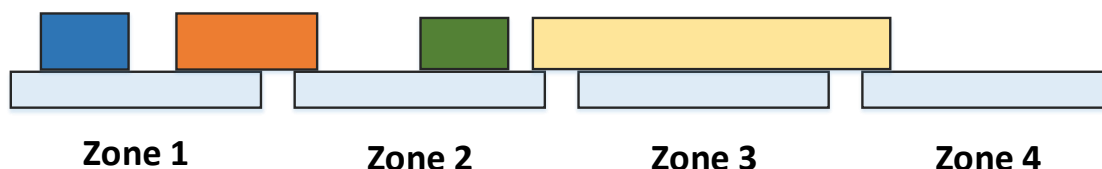
При ускорение и забавяне на обекта този наблюдател има достатъчно информация да консумацията на ток от мотора [12] и изменението на скоростта, за да оцени инерцията и статичния товар, но при равномерно движение може да се направи оценка само на статичното триене, което е недостатъчно за надеждно определяне на наличието на обект. Затова периодично се предизвиква промяна на ефективното напрежение към мотора, за да се предизвикат промени в тока и евентуално скоростта и да може да бъде извършена оценка и на динамиката на товара.

Този алгоритъм работи добре само когато обектите имат тегло над някакъв минимум, който може да бъде надеждно оценен и тогава себестойността на конвейера може да бъде намалена с премахването на фотосензора.

В случаите, когато обектите са метални е възможно използването на индуктивни датчици за наличие на обект, което освен с по-ниска себестойност, може да допринесе и с по-голяма надеждност.

Размера на зоната (дължината ѝ) на един конвейер се оразмерява по възможно най-дългия обект, който може да бъде придвижван. Полезна информация, която е налична в контролерите за управление на моторизирани ролери е текущата позиция на вала на мотора. Разполагайки с нея и информацията от сензори за наличие на обект, зоните могат да разположат много по-гъвкаво обектите върху тях, като максимизират

показателя за плътност на обектите Фиг. 8. Това позволява да бъдат придвижвани обекти по-дълги от дължината на зоната. А малките обекти (с малка дължина) могат да бъдат разположени по няколко върху една зона. Примери за такива конвейерни системи могат да се намерят в сортировъчните центрове на куриерските и пощенските служби.



Фиг. 8 Разположение на обектите спрямо зоните

Показателите плътност на обектите и средна скорост на придвижване не могат да бъдат оптимизирани заедно, защото ако зоната се опитва да уплътни разстоянието на някой обект със следващият, тя трябва да спре за момент придвижването на първия, докато празното разстояние до следващия се намали достатъчно. Това влошава другия показател за средна скорост на придвижване.

Създаването на алгоритми, оптимизиращи плътността на обектите върху конвейера или средната им скорост на придвижване, използвайки статистическа информация за поведението на системата и текущите ѝ параметри представлява интересна сфера за проучване и е в основата на настоящата работа.

1.6 Разпределени системи

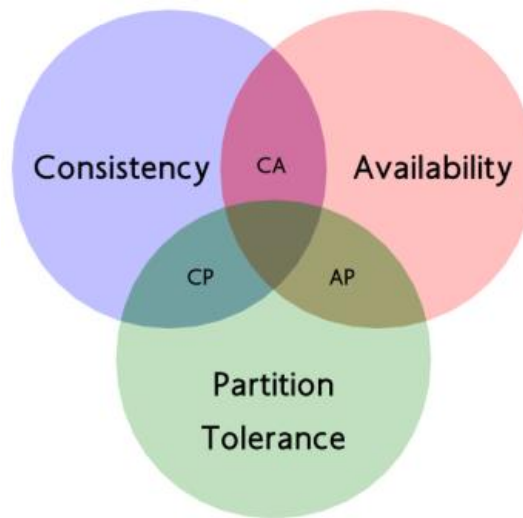
При разпределената обработка данните са разположени в отделни компютърни системи, което дава съществени предимства, но създава и нови предизвикателства.

Основните предимства се определят от повишаването на производителността чрез паралелна обработка на големи обеми от данни, както и елиминирането на единичната точка на отказ.

Предизвикателствата са свързани основно с осигуряване на консистентност на данните и изобщо със синхронизацията [13].

Напоследък все по-често се говори за големи данни, а те обикновено са разпределени, поради което изискванията към тяхната обработка са сходни с тези при разпределените данни.

Особеностите на разпределената обработка могат да бъдат представени и анализирани чрез теоремата CAP (Consistency, Availability, Partition tolerance) [14].



Фиг. 9 CAP теорема

Тя показва, че всяка разпределена информационна система (колекция от взаимосвързани изчислителни възли) може да поддържа едновременно само 2 от следните 3 свойства: съгласуваност, наличност и разпределеност.

Съгласуваност (Consistency) – Това свойство е основополагащо за информационни системи, тъй като е свързано с коректността на данните. Например, трябва да се осигури, че при четене от системата се връща най-актуалният запис.

По-конкретно, ако потребител обнови данните върху сървър1, то промяната ще се отрази и върху сървър2, който се намира в друг регион и потребителите на сървър2 ще прочетат същите данни, написани на сървър1.

Наличност(Availability) - Всеки мрежови възел (който не е паднал) връща отговор за всички прочетени и написани заявки за даден период от време. За да запази свойството наличност, той трябва да отговори в определен период от време, което зависи от приложението и се нарича реално време.

Например: Ако даден потребител се опитва да прочете данни, те ще бъдат винаги налични, т.е. не се заключват с цел осигуряване на консистентност.

Разпределеност (Partition tolerance) - Системата продължава да функционира макар да е разпределена.

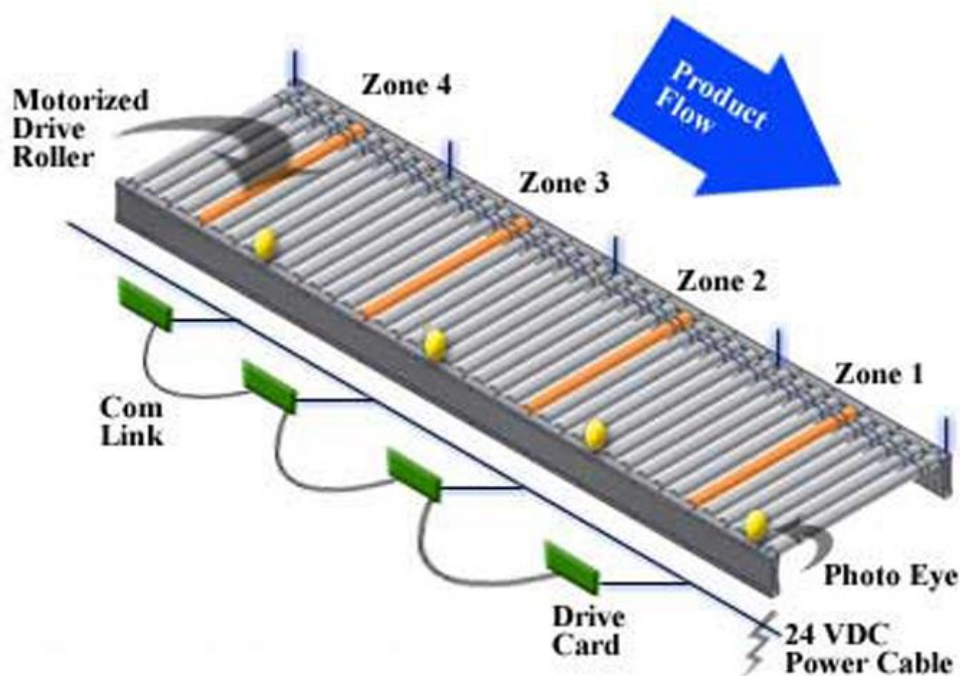
Разпределените системи за обработка на данни трябва да бъдат така проектирани, че да преминават за кратко време от състояние на

консистентност към състояние на достъпност и обратно, при което у потребителя остава впечатлението, че системата удовлетворява и трите изисквания едновременно.

1.7 Проблеми при захранването на конвейерни системи с МР

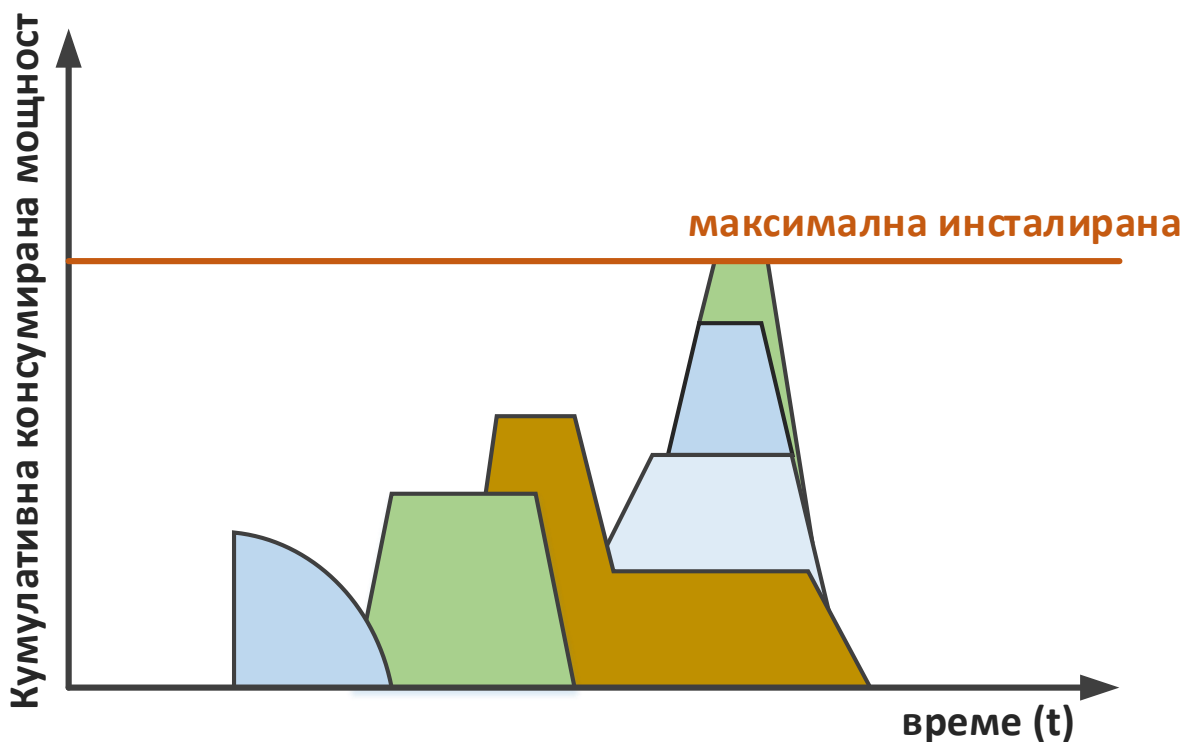
1.7.1 Инсталирана мощност

Захранването на конвейера се осъществява от захранващи блокове, които захранват по няколко зони и обикновено отделен блок, който захранва оперативното (логическо) захранване. Консумацията на оперативно захранване не се променя по време на работата на конвейера и не създава проблеми. Силовото захранване, обаче, което захранва моторите отправя някои предизвикателства. Съществува и вариант, в който няма отделен блок за логическо захранване Фиг. 10.



Фиг. 10 Захранване на конвейера

Оптимизирането на себестойността на съвременните конвейерни системи води до използването на захранващи блокове, чиято сумарна мощност е значително по-малка от сумарната мощност на инсталираните моторизирани ролери [13]. Като основен довод за това са проучвания, сочещи, че ако се инсталира захранване можещо да поеме цялата максимална моментна мощност на всички зони, то ресурса остава неизползван почти през цялото време на работа на конвейера.



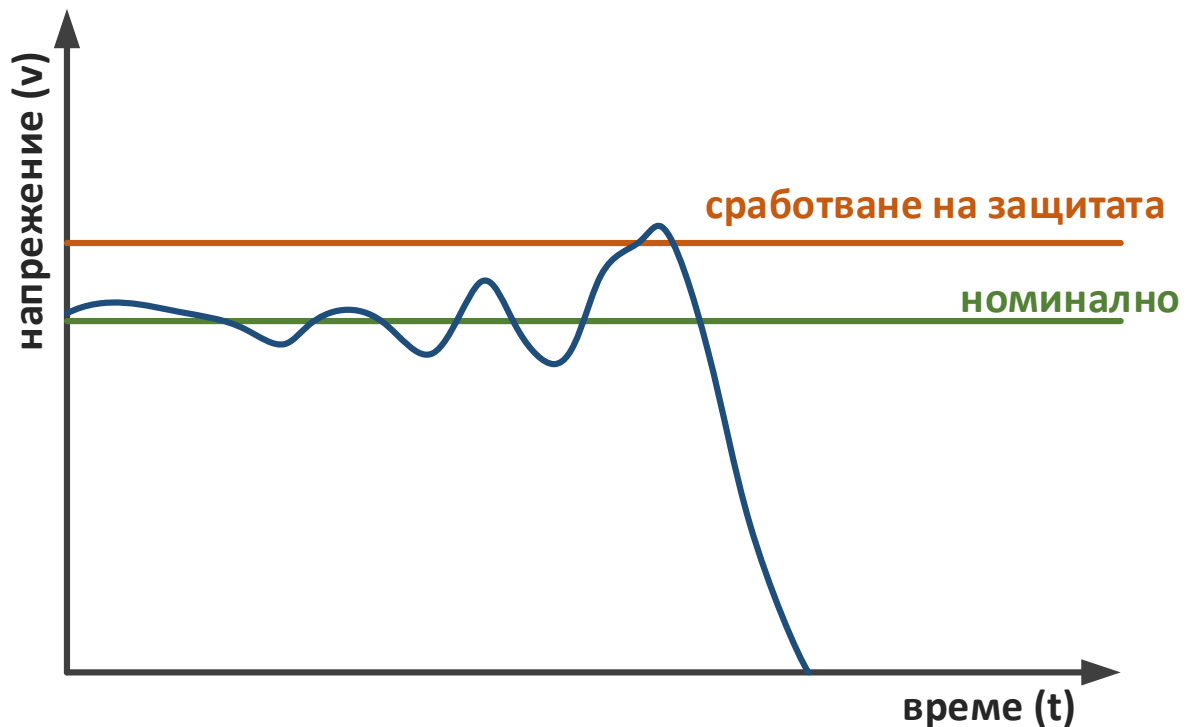
Фиг. 11 Примерна консумирана мощност в течение на времето

Инсталирането на захранване с по-малка мощност обаче води до проблеми, свързани със сработването на защитите на контролерите от понижено напрежение или защитите на блоковете от превишен ток [14], ако не бъдат взети мерки на софтуерно ниво. Обикновено такива мерки са удължаване на времената за ускорение на обектите и ограничаване на максималния ток през моторите. На Фиг. 11 е показан пример при който 4 зони, чиято консумирана мощност е дадена с различни цветове достига максимално инсталираната за определен момент във времето.

Всичко това води до подобряване на себестойността на системата, но за сметка на нейните характеристики, като средна пропускателна способност време за добиване на готовност за приемане на обект и време за достигане на крайната скорост на придвижване.

1.7.2 Върната енергия

Друг проблем свързан с електрозахранването на конвейерните системи е връщането на енергия към захранващите блокове при спиране или забавяне на движението на обектите. В най-честия случай захранващите блокове не могат да поемат външна енергия и да я преобразуват в електрическа и да я върнат към мрежата или да я разсейват [15]. Тогава могат да се получат сработвания на защитите от пренапрежение, ако в този момент няма зона, която да консумира върнатата енергия и мощността на оперативното захранване е по-малка от връщаната.



Фиг. 12 Примерно сработване на защита от пренапрежение

На Фиг. 12 е показан случай на сработване на защитата от пренапрежение. Примерни стойности на праговете за реална система за $U_{\text{ном}} = 24V$ и $U_{\text{пренапрежение}} = 30V$.

Ефекта на повишаване на захранващото напрежение се засилва при обекти с по-голямо тегло, заради повечето натрупана в тях кинетична енергия.

Друг нежелателен ефект може да бъде повредата на електрооборудване захранено от същия захранващ блок, както и на самия блок, ако не е предвидена защита от пренапрежение. За повечето устройства консумиращи енергия защитата представлява т.н. Transient Voltage Suppressor или TVS – компонент, който разсейва мощност в топлина, опитвайки се да поддържа напрежението под определена стойност. В повечето случаи тази защита не може да бъде правилно оразмерена от производителите, защото не е налична предварителна информация за количеството мощност, която трябва да бъде разсеяна и ролята на защитата се изразява в индикация за повреда от пренапрежение (при нейното дефектиране).

Връщането на енергия може да бъде неутрализирано в контролера на зоната, чрез компенсираща верига, която разсейва излишната мощност като топлина.

1.8 Проблеми с надеждността на данните в силно зашумена среда

При комутацията на БЧПТ двигатели се използва информацията от сензори за положението на ротора на мотора [16], както беше обяснено в Ш.5.2. Тези сензори са разположени вътре в мотора обикновено монтирани на печатна платка Фиг. 13 и сигналите от тях трябва да бъдат предадени по кабел Фиг. 55 до контролера, който да извърши подходящата комутация в зависимост от сигналите от тях [17].



Фиг. 13 Датчици на Хол монтирани на печатна платка

Предаването на данните може да е директно, като сигнали от датчиците по отделни проводници или кодирано по един проводник [18]. С течение на времето вторият начин набира популярност и разпространение, защото се намалява общото сечение на кабела от ролера. И в двата случая обаче електромагнитните смущения от комутация на намотките може да доведе до грешка в получената от контролера информация, защото проводниците се намират в един кабел и имат силно капацитивно обвързване.

1.9 Проблеми свързани със защитата на интелектуалната собственост вложена в модулите за управление

Сигурността е динамична сфера изискваща непрекъснато развитие и подобрения на всяко ниво, за да може да се постигне защита срещу различни видове възможни атаки.

Много индустриални системи, сред които са и конвейерите представляват своеобразен екземпляр от така неречените Интернет на нещата IoT технологии, защото обикновено в структурата си включват маршрутизатори, които позволяват връзка до други устройства през тях [19]. Докато преноса на чувствителни данни през такава инфраструктура,

обикновено има лесно решение с криптиране в двата края на връзката, сигурността на работещия персонал свързваха със здравето му си има своите решения, които също са добре разработени, както и сигурността на самото изделие или цел на индустриалната система има друг проблем, свързан със сигурността, който често е пренебрегван. Това е възможността за клониране и пресъздаване на самата система или части от нея Фиг. 14, в която производителя е вложил индиректни инвестиции под формата на проучвания, разработвания на хардуер, софтуер и алгоритми.



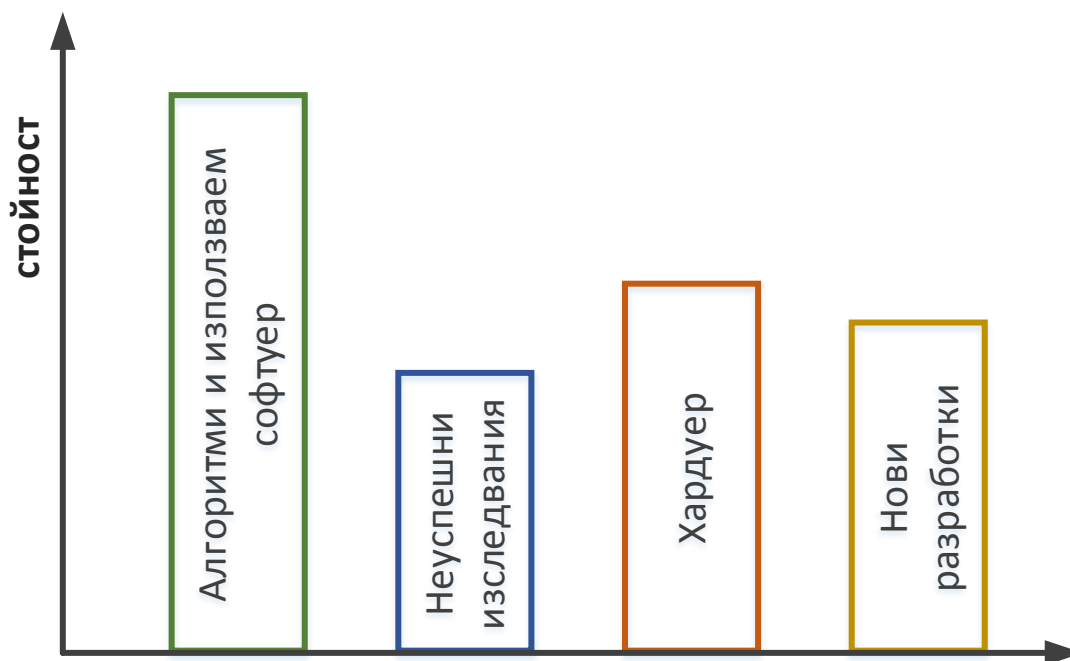
Фиг. 14 Клонирано и оригинално устройство

Използването на копия на оригинални компоненти в системата може да доведе до функционалното компрометиране на цялата система, както и до влошаване на нейните характеристики, до злополуки с работещия персонал и повреди на скъпо оборудване, до загуба на данни.

Патентоването на възможно повече компоненти от една система дава някаква сигурност, но представлява голям разход, особено в случаите на международен патент, а и при наличие на богата гама от устройства, които не могат да бъдат защитени с един патент. Освен това патентоването на софтуер е невъзможно. Той трябва да бъде обособен и формулиран като метод, способ и т.н., което е не винаги възможно, защото едни и същи методи се ползват в различни сфери и най-вероятно са вече патентовани.

Освен това малки промени в алгоритмите могат да използването на почти същия алгоритъм без да попада в сферата на подобен патентован.

Пример за такова копиране е създаване на хардуерно копие на контролер, управляващ зона на конвейер, където да бъде записан изпълнимия код след прочитането му от оригиналния контролер. Инвестицията на компания, която би извършила това е само за разработването на копие на хардуера (печатна платка, кутия и външно оформление) и съвсем малко за четенето и записването на софтуера.

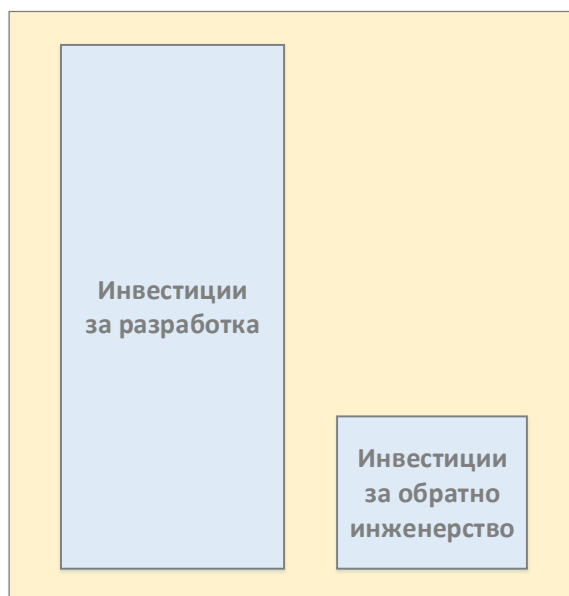


Фиг. 15 Инвестиции на производителя на контролера

Компанията разработила контролера обикновено има многократно по-големи инвестиции за разработването на софтуера, в сравнение с хардуера Фиг. 15. Не малка част от разработките са неуспешни, но са отнели ресурс, който тя трябва да възстанови от някъде. Ако компанията се опита да възвърне тези инвестиции с малко количество, продадени в началото контролери те биха станали неконкурентноспособни и непродаваеми, заради високата си цена, затова компаниите разчитат на продажбата на достатъчно голям брой контролери с малка добавена стойност. Друг аспект, на възвръщаемостта на средствата е те да бъдат използвани за нови проучвания и допълнително развиване на продукта, като също част от тези опити понякога се оказват неуспешни или пазарът не е готов за тях.

От друга страна за конкурентна компания може да се окаже по-малка инвестиция обратното инженерство [20] и дублирането на хардуера, както и прочитането на софтуера от готов контролер, отколкото да създава двете на ново Фиг. 16. Така копието на продукта може да бъде предложено на

много по-ниска цена и да завземе значителна част от пазара, отнемайки възможността на оригиналния производител да възвърне средствата си. Това като следствие води до застои в развитието на изделието.



Фиг. 16 Сравнение на инвестициите за разработване и обратно инженерство

За да се предотврати това трябва производителя да така да затрудни и оскъпи процедурата по обратно инженерство, че тя да се изравни или надмине собствените ѝ разходи за разработване. Това би обезсмислило клонирането на контролери.

1.10 Софтуерни агенти

Компютърните програми стават все по-интелигентни и способни да вземат самостоятелни решения, което се свързва с понятието софтуерни агенти.

Съществуват много примери за софтуерните агенти. За да поясним техните характеристики ще използваме пример със система от светофари. Ако светофарите се управляват централизирано, то всяко кръстовище е просто изпълнител на команди, които се подават от центъра за управление. С добавяне на интелигентност обаче, всяко кръстовище би могло да решава самостоятелно, колко автомобили да пропуска. За тази цел отделните кръстовища трябва да обменят информация по между си.

В подобна децентрализирана система всяко кръстовище е отделен агент, който има свои цели (например постигане на най-малко задръстване) и предприема самостоятелно действия за осъществяването им.

Софтуерните агенти имат собствени цели (проактивност), постигат ги автономно (автономност) взаимодействайки със средата (реактивност).

Най-общо казано, софтуерните агенти действат самостоятелно, т.е. са автономни. Софтуерните агенти (СА) намират широко приложение, например за събиране на информация с цел подпомагане вземането на решения.

Съществуват много видове мрежови агенти, които изпълняват различни функции, като например търсене на пробиви в сигурността, проверка за повреди и др.

Съществуват алгоритми, които са базирани на груповото поведение на мравките, пчелите и др. Идеята е, че чрез използването на многоагентна система (МАС) може да се симулира много сложно поведение, дори всеки отделен СА да има проста функция.

СА могат да се използват широко в компютърните симулации, тъй като с използването на МАС може да се постигне сложна функционалност. Например за симулация на автоматизиран склад, където отделните участници се представят като СА и изпълняват своята роля. Да произвеждат стоки, да ги транспортират, да търсят и т.н.

СА намират широко приложение в компютърните игри, като симулират сложно поведения. Не отдавна бе оповестен случай, при който СА представляващ участник/противник в играта, успява да победи абсолютни всички останали играчи. В случая СА е постигнал напълно своята цел, но играта губи потребители, тъй като много от тях се отказват.

1.11 Изкуствен интелект и машинно обучение

Изкуственият интелект е свързан с проявата на интелигентно поведение. За разлика от човешкия интелект обаче, изкуственият се осъществява в неща, които са създадени изкуствено, като например компютърни програми.

Терминът се използва за да опише компютърни програми, които наподобяват функциите на човешкия мозък, като възможност за заучаване на нов материал и решаване на сложни проблеми.

Въпросът, до колко компютърните програми могат заменят човешкия интелект има дълга история, но е факт, че машините са способни да извършват действия изискващи интелигентност.

Стратегическите игри, като например шах, несъмнено изискват сериозни мисловни процеси, но дори в тази задача компютрите надминават и най-добрите шахматисти.

Несъмнено „мисленето“ на компютрите е различно от това на хората, като в някои отношения е много по-ефективно.

Дълги години хората са се опитвали безуспешно да полетят имитирайки полета на птиците, докато не са открили по-ефективен начин. Съвременните самолети не летят като птиците, но го правят много по-високо и по-бързо.

Подобна аналогия може да се направи и за изкуствения интелект. Той не е като човешкия, но за някои задачи се справя значително по-добре.

Чрез методите на машинното обучение например могат да се създадат програми, които откриват шаблони в данни. Първоначално се обучават с използване на тестови данни, а след това биха могли с висока степен на вероятност да предскажат определени събития. Например, въз основа на тестови данни за поведението на потребителя, програмата би могла да предскаже, дали той би закупил дадена стока.

Машинното обучение е основен раздел на изкуствения интелект, който намира огромно приложение в обработката на данни, тъй като позволява на компютърните системи да се обучават автоматично и да намират шаблони в данните, за които не са изрично програмирани.

Чрез методите за машинно обучение, от данните може да се извлече знание. Например въз основа на медицински данни може да се предскаже, дали даден пациент попада в определена рискова група и съответно да се предприемат превантивни действия.



Методите за машинно обучение обикновено се състоят от две фази:

Фаза на обучение при която въз основа на тренировъчни данни се създава модел;

Фаза през която се извършва същинската обработка, като например класификация на данните.

През фазата за обучение автоматично се създава модел, въз основа на тренировъчните данни, което определя и името на цялото научно направление, а именно машинно обучение.

Съществуват два основни класа методи (алгоритми) на машинното обучение: **Supervised** (контролирано) и **Unsupervised** (безконтролно).

Схематичното представяне на отделните методи на машинно обучение може да се изрази по следния начин:

При методите на контролираното обучение се използват тренировъчни данни, които са етикетирани. Към тях спадат регресионния анализ и класификацията. Регресионният анализ обикновено се използва за предсказване на една стойност, например как ще се промени цената на имотите в зависимост от квадратурата, докато при класификацията се определят много (клас от) стойности.

Към безконтролните методи се причислява клъстеризацията, където липсват етикетирани данни и предварително създаден модел. Моделът се създава в процеса на клъстеризацията, въз основа на данни, за които не се знае предварително към кой клас попадат.

За пояснение, ще използваме пример за класификация на прибори за хранене (вилници, лъжици и ножове). При контролираното обучение се използват тренировъчни данни, които са етикетирани. т.е. показва се лъжица, на която пише „Лъжица“. След това се показва вилица на която пише „Вилница“ и така постепенно системата се обучава да разпознава обектите, за които предварително се знае към кой клас се отнасят. След като системата се обучи ѝ се дават нови данни, които тя трябва сама да класифицира.

При безконтролните методи, фазата на обучение се осъществява без да са налични етикетирани данни. В този случай системата сама трябва да определи в колко класа да ги групира обектите според характеристиките им, т.е. да създаде съответния брой клъстери в които попадат отделните обекти.

Изкуственият интелект намира широко приложения [21]. Например в разпознаването на изображения, разпознаването на естествен език, в стратегически игри, при управлението на автономни автомобили и много др.

В термините на изкуствения интелект (ИИ) се среща понятието база знания. За разлика от базите данни, където се търси съществуващ запис, при базите знания може да се извлече знание, което не съществува в явен вид.

Често при конвейерните системи обектите се придвижват на случаен принцип. Ако системата изгради модел на поведение, то може да се опита да буферира като уплътнява по-добре зоната [22].

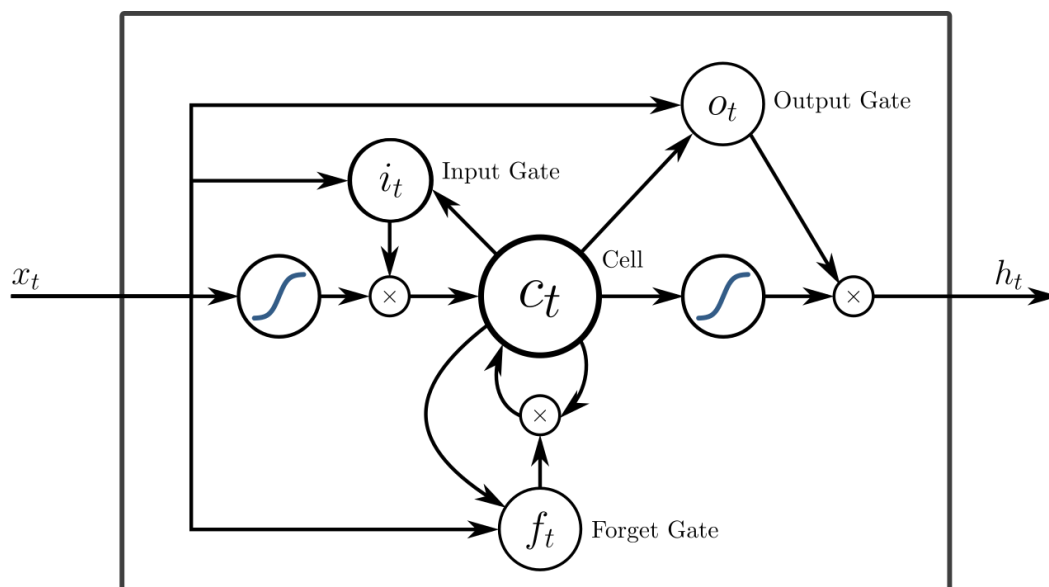
Съществуват много средства, чрез които бързо и лесно може да се имплементира система използваща машинно обучение, като например PyTorch, TensorFlow, Torch, Keras, scikit-learn и др., но без данни не може да се получи адекватно обучение.

1.11.1 Невронни мрежи

Невронните мрежи могат да се причислят към методите за машинно обучение, тъй като и при тях има две фази: трениране и използване, но имат и характерни особености.

В случая на конвейерни системи, при които средния брой обекти за единица време преминаващи през конвейера, общото количество обекти върху определена линейна част от него, отношението на броя навлизащи към броя напускащи участъка обекти и т.н. са величини с някаква периодичност във времето. Подходящи за моделиране на такъв вид величини са невронните мрежи от тип Long short-term memory [23], [24].

При тези мрежи единиците са със структура, показана на Фиг. 17 и съдържат вход и изход, както и забравящ възел, който може да бъде разглеждан, като стандартен неврон от Feed-forward или многослойна невронна мрежа.

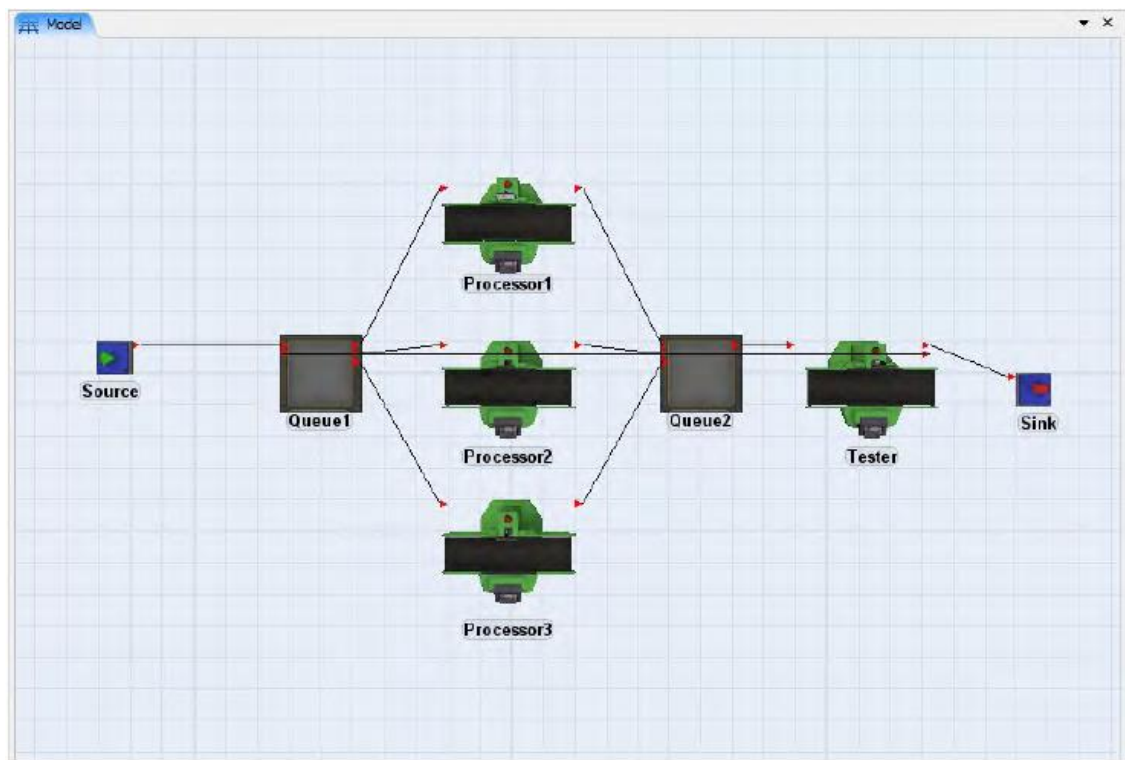


Фиг. 17 Структура на единиците на LSTM невронна мрежа

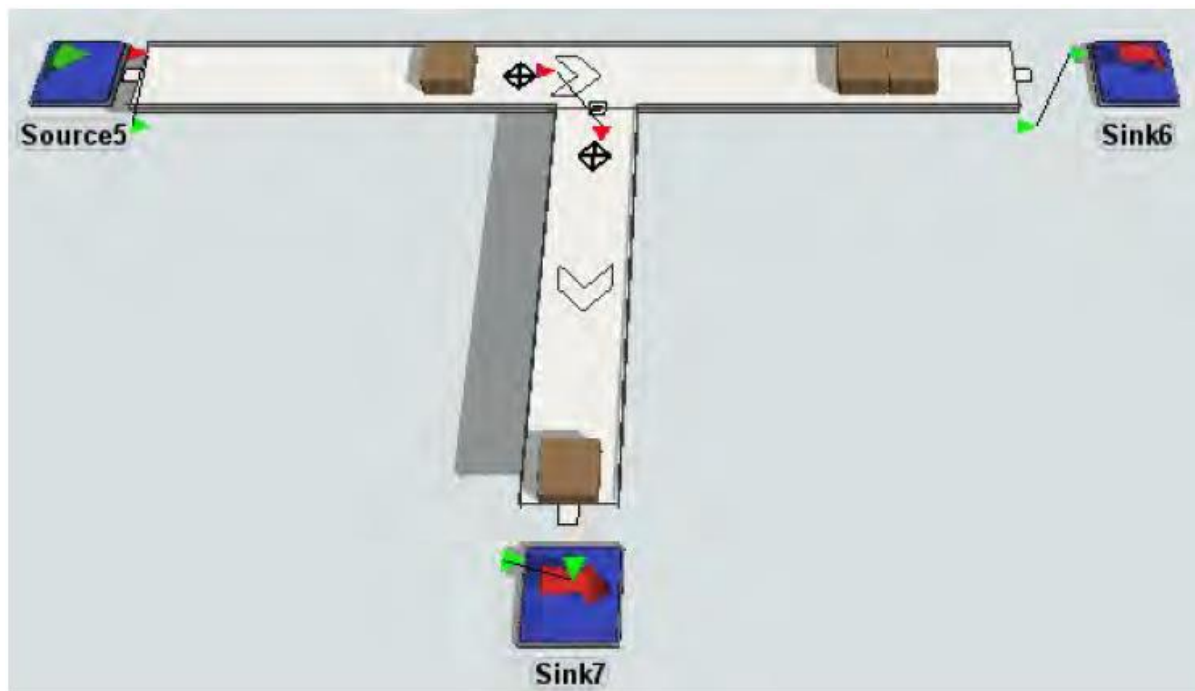
1.12 Компютърни симулации

Подходящи за компютърни симулации [25], [26] са софтуерни продукти работещи в дискретно време, базирани на събития от симулирания процес, като FlexSim и SimCAD. Първият е по-удобен, заради

приятелския интерфейс и лесния начин за създаване на модели Фиг. 18 и Фиг. 19.

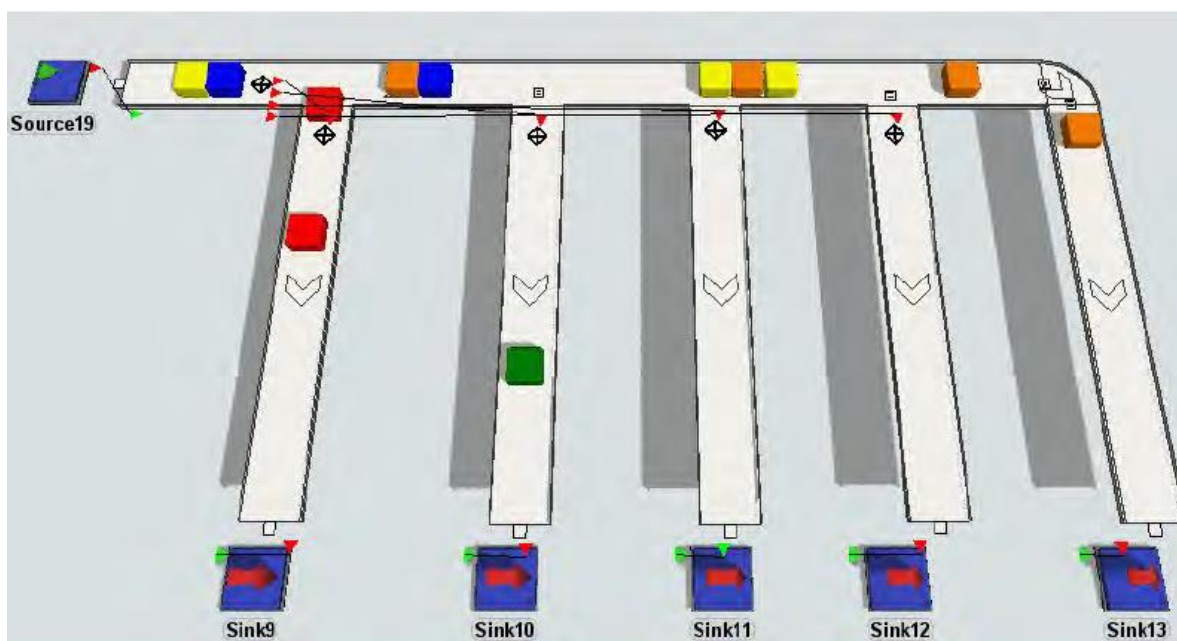


Фиг. 18 Създаване на модел в FlexSim



Фиг. 19 Симулиране на конвейерна система с кръстовище

Софтуерът предлага 3Д визуализация на процеса, при който времевия мащаб може да бъде променен. Така симулация, която трябва да се извърши за часове, може да бъде симулирана и прегледана за няколко секунди [27], [28].



Фиг. 20 Симулиране на сортировъчен център.

1.13 Заключение

Модерните контролери за управление на конвейери предоставят много източници на информация за придвижваните обекти, като разстояние на придвижване на моторизирания ролер. Те също така имат много по-голяма изчислителна мощ спрямо предишното поколение.

Използването на максимума от наличната информация в контролера разкрива нови възможности за създаване на по-ефективни алгоритми повишаващи качеството на придвижване на обектите.

Необходимо е да се предложат нови методи за:

- Интелигентно разпределено управление;
- Ефективно разпределение на енергията между зоните;
- Защита на интелектуалната собственост.

I.14 ЦЕЛ И ЗАДАЧИ НА ДИСЕРТАЦИОННИЯ ТРУД

Основната целта на дисертационния труд да се предложи решение за разпределено управление на конвейерни системи базирано на софтуерни агенти, което чрез използване на наличната информация за товарите пътуващи върху конвейера подобрява глобалните характеристики на системата.

Поставени са следните допълнителни цели:

1. Да се верифицира предложеният метод за разпределено управление, чрез използване на симулационен софтуер.
2. Да се проектира и имплементира метод за установяване и разпределение на консумираната енергия от захранващите източници.
3. Да се предложи и имплементира метод за повишаване на надеждността при предаване на данните между електрическите мотори и контролера.
4. Да се предложи и имплементира метод за подsigуряване на контролерите срещу клониране.

II ГЛАВА. РЕШЕНИЯ ЗА ДОБАВЯНЕ НА ИНТЕЛИГЕНТНОСТ КЪМ СИСТЕМИТЕ ЗА УПРАВЛЕНИЕ НА КОНВЕЙРИ

В тази глава се предлагат няколко решения за подобряване характеристиките на конвейерната система, чрез добавяне на интелигентност, при която всяка зона се разглежда като самостоятелен агент, решаващ разпределено част от общата задача за постигане на целта.

II.1 Предложение за подобряване ефективността на конвейерни системи, чрез прилагане на алгоритъм разпределено управление

За всеки самостоятелен агент, ще бъде дефинирана целева стойност на важен показател от конвейера, като изчислената стойност на този показател ще бъде различна в зависимост от относителното положение на зоната (агента) по дължината на линейния участък на конвейера.

II.1.1 Архитектура на предложената разпределената система за управление

Интелигентен начин за управление на зона на конвейер може да се разработи, който работи ефективно както при къси, така и при дълги обекти. Разпределения алгоритъм може да опитва да акумулира колкото се може повече обекти на конвейера, като запазва средната скорост на придвижване висока. За да се получи този начин на управление трябва да има комуникация между зоните и информация за разстоянието което моторизирания ролер е изминал.

Управлението на всяка зона ще бъде разгледано, като самостоятелен агент извършващ отделни самостоятелни пресмятания и изводи, който за краткост ще бъде наричан 'зоната'.

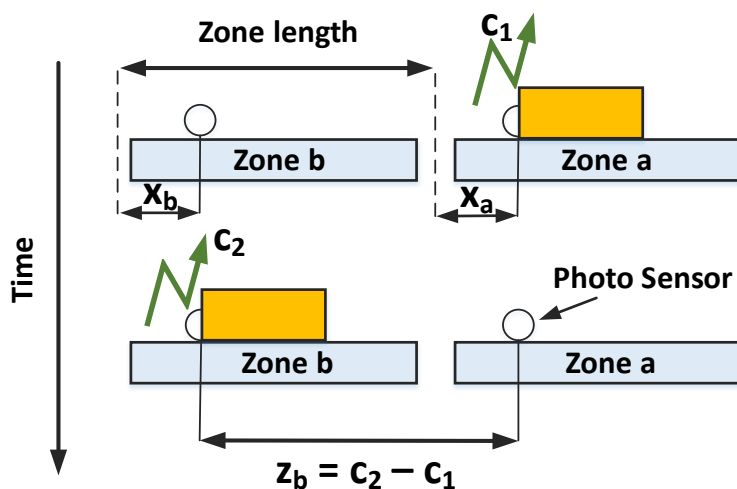
Дадена зона може при получаване на обект от предхождащата зона да запомни стойността на това разстояние като c_1 . Когато на сензорът ѝ се появи сигнал тя записва стойността на това разстояние като c_2 . Има и някакво разстояние между фотосензора на предхождащата зона и края ѝ, което ще бъде отбелязано с x_a и същото такова разстояние между фотосензора на разглежданата зона и нейния край, което ще бъде отбелязано с x_b . Така може да бъде записан израз (1) за дължина на разглежданата зона z_b :

$$z_b = c_2 - c_1 + x_b - x_a \quad (1)$$

Графично това е показано на Фиг. 21. Обикновено всички зони от линейната част на един конвейер са еднакви и може да се приеме, че разстоянието между фотосензора и края на зоната е еднакво. Тогава могат да бъдат изведени (2) и (3):

$$x_a = x_b = x \quad (2)$$

$$z_b = c_2 - c_1 \quad (3)$$



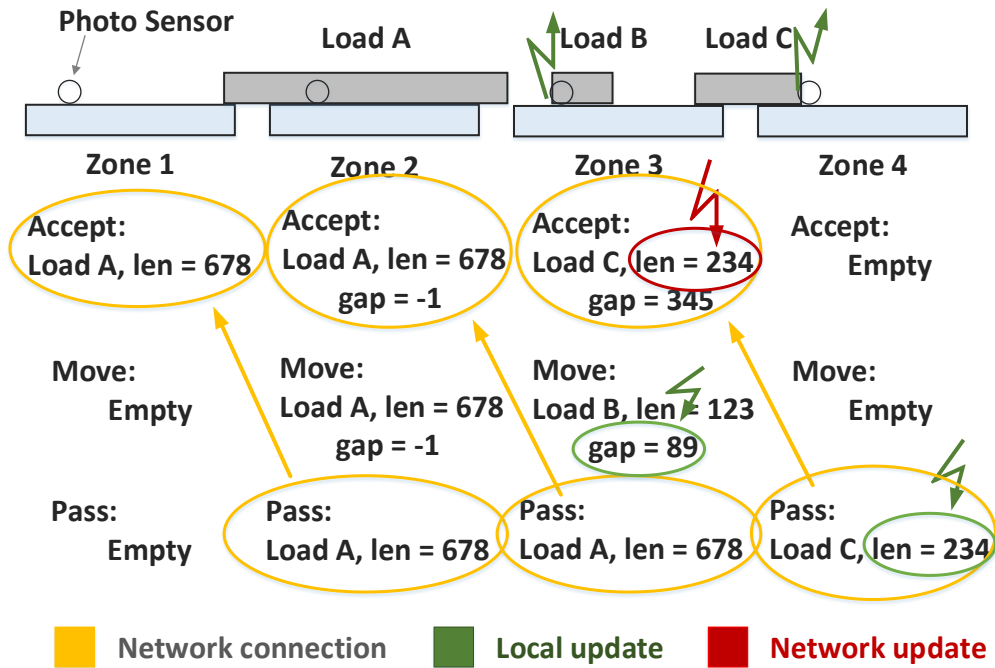
Фиг. 21 Измерване на дължината на зоната

Зоната може да измери дължината на обекта по разстоянието, което нейният МР е изминал между събитията, когато обекта се е появил c_2 и когато е изчезнал c_3 от нейният фотосензор (4).

$$l_b = c_3 - c_2 \quad (4)$$

Ако зоната извърши подобно измерване, но на противоположните събития – когато обекта е изчезнал от фотосензора й c_3 и когато се е появил отново обект c_4 това ще бъде мярка за празното разстояние между току що заминалия обект и новопристигналия (5).

$$g_b = c_4 - c_3 \quad (5)$$



Фиг. 22 Всяка зона запазваща три списъка с обекти

На Фиг. 22 е показано предложение, в което всяка зона поддържа три списъка с обекти:

- Които се приемат в момента с дължина 0 или 1 наречен Асепт.
- Които в момента се придвижват по зоната наречен Move, чиято дължина може да е между 0 и някаква максимална стойност, която може да бъде предварително конфигурирана – напр. 10
- Които се предават към следващата зона наречен Pass в момента с дължина 0 или 1.

В началото, когато някое разстояние не е известно още, неговата дължина се инициализира с -1.

На Фиг. 22 е показано как събитията от фотосензорите предизвикват обновяване на измерените стойности на празните разстояния между обектите и тяхната дължина на зони 3 и 4. В случая промяната на измерената дължина на обекта върху зона 4 предизвиква обновяване на информацията в зона 3, чрез предаване на данни по комуникационната мрежа.

С наличието на информация за дължината на обектите, на самата зона и на празното разстояние между обектите, зоната може динамично да променя празното разстояние, като подобрява средната скорост на придвижване или гъстотата на обектите.

Първата и последната зони могат да се определили като такива използвайки информация за разликата във функционалността на съседните

зони, предавана по мрежата. Използвайки тази информация всяка зона между тях може да определи относителното си положение изразено като разстояние спрямо началото и края на линейния участък.

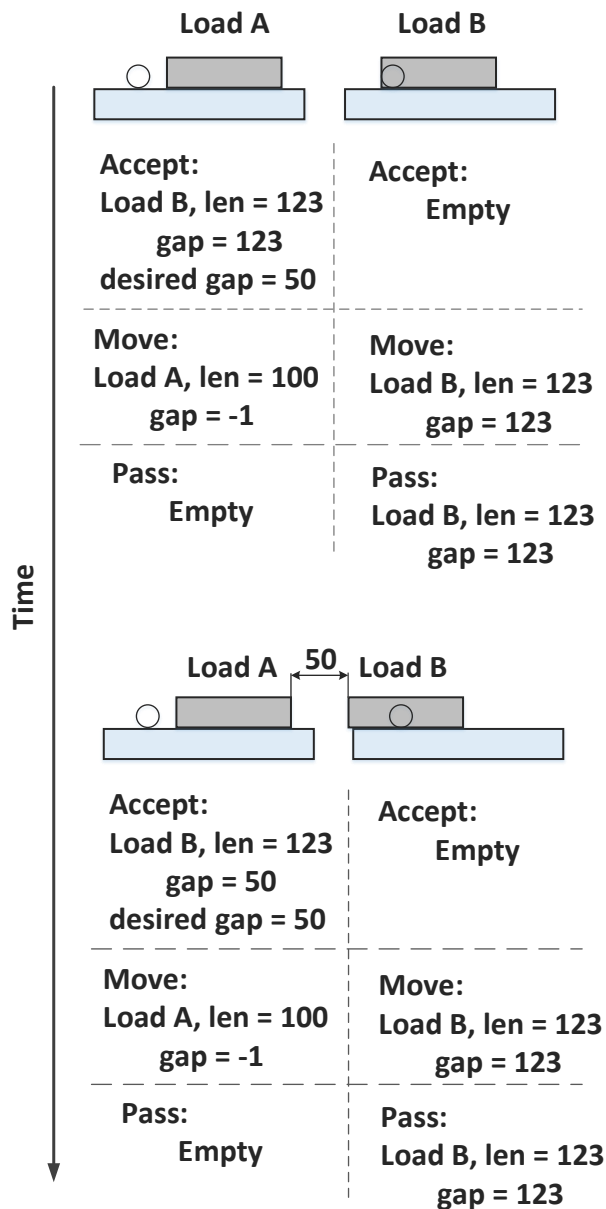
II.1.2 Предложение за алгоритъм за разпределено управление

Конвейера може да бъде разделен на линейни участъци и кръстовища, които ги свързват. За всеки обект върху линейния участък на конвейера, чрез предаване на информация по мрежата се изчисляват, колко броя обекти има приети от началото до него и колко има от него до края на участъка, чакащи да бъдат предадени.

Използвайки тези данни се изчислява оптимално празно разстояние така, че колкото повече обекти има след текущия, толкова по-малко става това разстояние. Това пресмятане се извършва разпределено от всяка зона, като зависимостта на изчисленото празно разстояние може да е линейно или с друга зависимост по отношение на относителното разстояние на зоната по дължина на конвейера.

Предварително е конфигурирано минимално празно разстояние под, чиято стойност не трябва да пада резултата от тези изчисления с цел фотосензорите да могат надеждно да отчитат наличието на празното разстояние между два съседни обекта. Конфигуриран е и максимален брой обекти, над-който изчисленото празно разстояние да е винаги минималното, с цел да се даде възможност за увеличаване на плътността на броя акумулирани обекти.

Когато дадена зона определи, че изчисленото празно разстояние между два обекта е по-малко в сравнение с действителното, което е получено като информация от предишната зона тя може да го намали, чрез намаляване на скоростта или спиране за момент. Това се извършва по време на приемане на обекта и докато обекта все още не е навлезнал в рамките на зоната, както е показано на Фиг. 23. След като обекта е вече едновременно върху двете зони те се движат с еднаква скорост, за да няма приплъзване на обекта, което е недопустимо.



Фиг. 23 Скъсяване на празното разстояние по време на приемане на обект

II.1.3 Разпределено изчисляване на целевата функция

Ако означим даден обект от линеен участък на конвейера с j , плътността на обектите върху линията може да бъде намерена, като отношението на сумата от дължините на обектите до началото и сумата на дължините на зоните. Дължините на зоните обаче включват компонента, която трябва да бъде конфигурирана, а именно дължината от фотосензора до дължината на самата зона. Затова не е желателно да се използва.

Друг метод за изчисление на тези плътности е да се изчисли отношението на сумата от дължините на обектите към сумата от дължините на празните разстояния между тях. Нека отбележим плътността от обекта j

до началото на линейния участък с \overleftarrow{d}_j , а на тези от обект j до края му с \overrightarrow{d}_j . Тогава за плътността преди обект j ще получим (6).

$$\overleftarrow{d}_j = \frac{\sum_{i=0}^j l_i}{\sum_{i=0}^j g_i} \quad (6)$$

А за тази след обекта j (7).

$$\overrightarrow{d}_j = \frac{\sum_{i=j}^n l_i}{\sum_{i=j}^n g_i} \quad (7)$$

Използвайки плътностите на обекти изчислени за текущата зона j може да се използва целева функция за желаното празно разстояние отбелязано с \hat{g} между обектите от вида (8):

$$\hat{g}_j = k \frac{\overleftarrow{d}_j}{\overrightarrow{d}_j} \quad (8)$$

Където k е коефициент, който се конфигурира и определя средния размер на празното разстояние и разпределението му по дължина на линейния участък.

Всяка зона преизчислява тази стойност за всички обекти в списъка Move когато получи нова информация от предишната или следващата зона. За да де минимизира трафика по мрежата всяка зона изпраща получената до момента сума на дължини на обекти и дължини на празни разстояния, като добавя тези от собствения списък Move. Така се получава разпределено изчисление на сумите от (6) и (7).

II.1.4 Реакция при неочаквани събития

Всяка зона извършва измервания на дължината на всеки преминал обект и празното разстояние до следващия и осреднява стойностите в зависимост от позицията си от началото със стойностите получени от предишните зони. Определя се статистическата грешка за дължината на обектите ε_{lj} по (9) [29].

$$\varepsilon_{lj} = \frac{|\widehat{l}_n - \bar{l}_n| + \varepsilon_{lj-1}}{2} \quad (9)$$

Където:

\widehat{l}_n е измерената дължина на обект n от зона j .

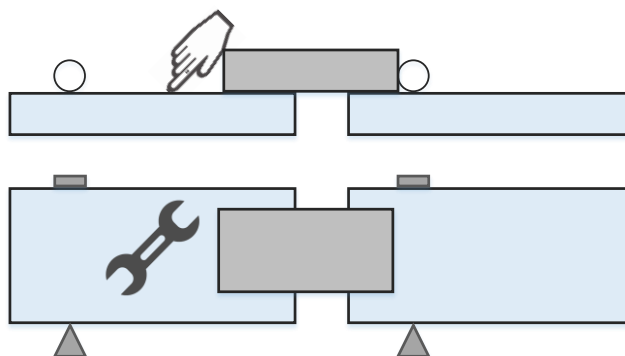
\bar{l}_n е получената дължина за обект n от зона $j-1$.

Аналогично грешката за празните разстояния се изчислява по (10).

$$\varepsilon_{gj} = \frac{|\widehat{g}_n - \bar{g}_n| + \varepsilon_{gj-1}}{2} \quad (10)$$

Двете грешки заедно с усреднените стойности на дължините се предават към следващата зона. Така с придвижването си надолу по линията обектите и разстоянията между тях добиват все по-точно измерени стойности или т.н. oversampling [30].

Ако зоната открие, че измерена стойност на дължина или обект е значително по-различна от получената чрез сравняване на $|\widehat{g}_n - \overline{g}_n|$ и ε_{gj-1} например за празно разстояние тя може да приеме, че има приплъзване или някаква външна интервенция спрямо обекта както е показано на Фиг. 24.



Фиг. 24 Неочаквано събитие

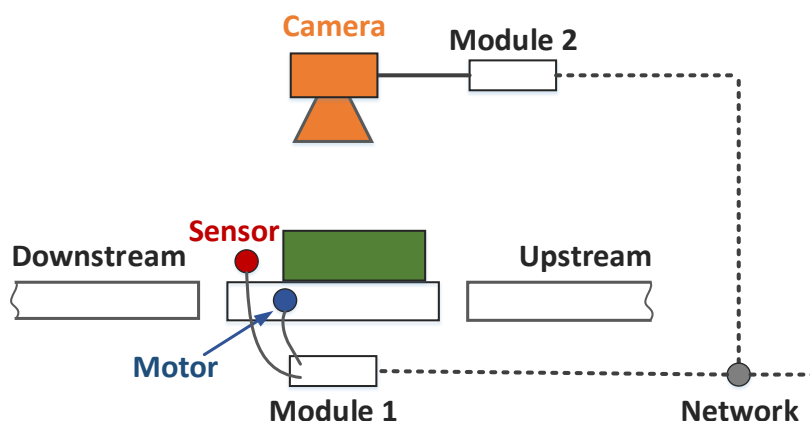
В такъв случай зоната не взема под внимание измерената от нея стойност, а използва предадената ѝ преди и създава сигнал достъпен за четене от софтуера на по-горно ниво, ако има такъв. Това обикновено е операторска станция, която показва състоянието на системата с подходящ интерфейс.

Не са редки случаите в практиката, в които ускоренията и забавянията с които са конфигурирани зоните са твърде бързи и предизвикват приплъзвания на обектите при тръгване и спиране. Друга причина е намаляването на сцеплението между обекта и зоната с времето поради зацапване, изхабяване на повърхностите и т.н. Затова, като допълнителна функционалност е предложено да се реализира брояч на събитията, в които са получавани големи грешки от измерване на дължини, който да бъде намаляван с единица през голям конфигуриран интервал от време. Ако неговата стойност достигне някаква прагова стойност, която също може да се конфигурира, да се извърши намаляване на стойностите на ускорението и забавянето на зоната.

Такава адаптация на стойностите на ускорение и забавяне е много ефективна на места, където преминават леки пластмасови и/или хартиени обекти, които имат лоша адхезия с повърхността на зоната, която в най-лошия случай е металната тръба на ролера.

II.1.5 Допълнителен модул за разпознаване на изображение и измерване

Като допълнение за начално измерване на размерите на обектите може да се използва (Charge-Coupled Device) CCD камера и лазерен излъчвател имплементиращи т.н. Time of Flight тип измерване показани на Фиг. 25. Времето за снимане и измерване е синхронизирано с движението на зоната по комуникационната мрежа.



Фиг. 25 Допълнителен модул за измерване на размери използващ камера

Като ценово по-ефективен метод може да бъде използвана и обикновена камера, която да търси разлики между две последователни заснемания по време на движение на зоната. С подходящ алгоритъм се търсят пикселите в които се наблюдават разлики и се определя размера на диапазона, който обхващат. Методът работи добре, когато на изображението се направи математическа обработка с матрица, коригираща изкривяването на оптиката в обектива. Тази матрица се предварително се съставя, като се направят снимки на калибрационно изображение – напр. шахматна дъска с точен размер на квадратите. Използват се и някои техники от обработка на изображения за намаляване на ефектите на треперене, които може да са налични и в самата камера и на части от конвейера.

Този модул дава бързи резултати за дължината на обекта и празното разстояние и посредством комуникация с контролера на зоната тези данни могат да бъдат правилно обвързани с придвижвания обект, който, както беше споменато по-рано обикновено има идентичност под формата на уникален номер пътуващ с него. Наличието на размера на обекта перпендикулярен на посоката на движение дава възможност на зоните управляващи перпендикулярни кръстовища като показаните на Фиг. 51 и Фиг. 52 да започнат да предават тези стойности на следващите зони в перпендикулярните направления.

II.2 Предложение за разпределяне на електроенергията между зоните

Предложеният метод позволява използването на захранващи блокове с много по-малки мощности в сравнение със сумарния на всички инсталирани модули за управление, което значително намалява себестойността на конвейерната система и премахва риска от задействане на защитите от пренапрежение и превишен ток [31].

Обикновено 100% от върнатата енергия спиране на моторизираните ролери се преобразува в топлина вътре в контролерите за управление. С предложения метод цялата или част от нея може да бъде използвана и така да се постигне много по-добра енергийна ефективност [32].

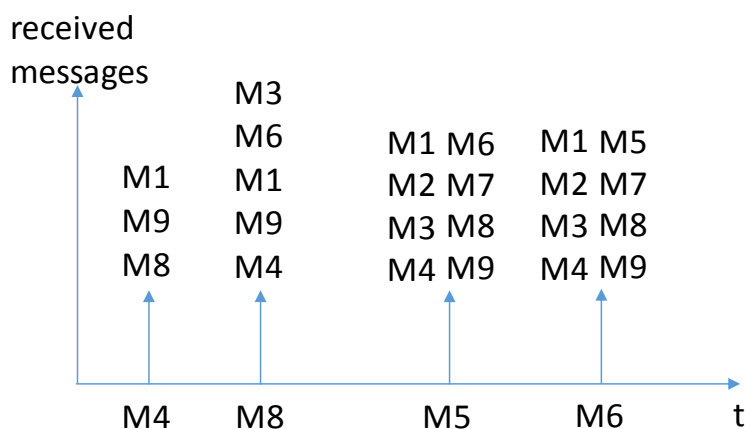
Решението се възползва от ползите на протокола за синхронизация IEEE 1588, който позволява на разпределени системи свързани по Ethernet да синхронизират вътрешните си часовници с точност до няколко микросекунди.

За изложението по-надолу ще бъде направено допускане, което е вярно при практическото изпълнение на системите, но за да е правилно теоретичния анализ то трябва да бъде изрично пояснено [33]: Захранващите проводници нямат паралелни връзки между които и да е две точки в системата. Това практически е така, заради повечето труд, който необходим за направата им и допълнителната цена за захранващ кабел.

Методът изисква постоянна консумация на ток във фазата на начално проучване, затова е удобно това да се извърши когато конвейера е захранен за първи път. В този момент моторизираните ролери още не се задвижват от алгоритмите за управление.

MAC address:

14:B1:26:00:00:00 - 14:B1:26:FF:FF:FF

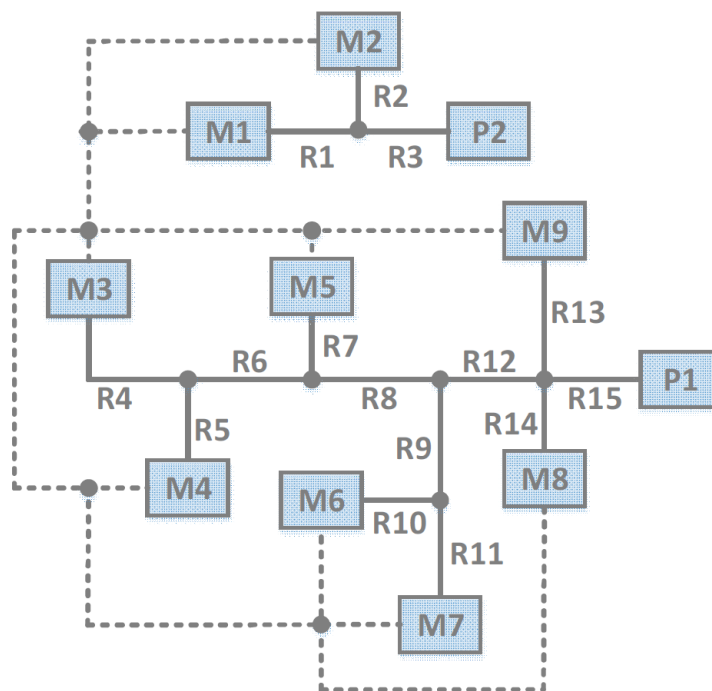


Фиг. 26 Времедиаграма на получените съобщения

След първоначално включване на електрозахранването модулите синхронизират вътрешните си часовници според протокол IEEE 1588 и всеки от тях започва слуша за broadcast съобщения (съобщение изпратено до всички) за покана за топографиране на захранването за известен период от време, конфигуриран в началото като 100ms. Ако не е получено такова модула изпраща такова съобщение с покана за започване на процедурата по топографиране. Ако модула получи такова съобщение в рамките на периода, той започва процедурата и не изпраща покана.

Ако модул получи съобщение за покана, за топографиране, но вече знае, че такова е извършено, изпраща отговор на подателя с данните от топографирането и че той може да използва само недеструктивен метод за намиране на собственото си положение на картата, което ще бъде разгледано по-късно.

Процедурата по картографиране започва, като всички модули изчисляват уникално число по MAC адресите на всички модули, от които са получили съобщения и го изпращат по мрежата. При получаване на такова съобщение и установено несъответствие в това число, всеки модул изпраща наново съобщение с показ за картографиране. Примерно съдържание на списъка с модули от които са получени съобщения е показано на Фиг. 26. Процедурата продължава, докато всички числа съвпадат, което означава, че всички модули имат един и същи списък със видими модули. Тази процедура е създадена, за да се реши проблема с различното време на стартиране на модулите при начално включване на системата.



Фиг. 27 Примерна схема за захранване и комуникация на конвейер

Нека разгледаме примерна конвейерна система показана на Фиг. 27. Прекъснатата линия показва комуникационния канал, а непрекъснатата хранващата линия. Блоковете с М са модули за управление на зона(и) на конвейер, а модулите с Р са хранващи блокове. С R са отбелязани съпротивленията на участъците от хранващия кабел.

Видимите един за друг модули, които има обща мрежа на комуникация и имащи вече еднакви списъци $L\{m_1, m_2, \dots, m_n\}$ със съседни модули го сортират по по-младшите 3 байта от MAC адреса. Това ще е редът, по който те ще консумират енергия за кратко време с техните компенсационни вериги, като периодът за това консумиране е предварително конфигуриран на 1ms.

В края на този кратък интервал модулът m_x , който е извършил натоварването изпраща broadcast съобщение по мрежата съдържащо времето от вътрешния му часовник на започване и свършване на това събитие, както и измереният консумиран ток и измерените хранващи напрежения преди натоварването V_0 и по време на натоварването V_L .

Другите модули очакват това съобщение и през това време извършват непрекъснато измерване на хранващото ги напрежение през интервал от 500us, като поддържат плаващ прозорец с 20 такива стойности. Периодът на измерване и размера на прозореца могат да се конфигурират.

След като получат съобщението те намират стойността от списъка, която отговаря на времето на натоварване в съобщението V_L , пресмятат средна стойност на всички останали измервания, което ще бъде отбелязано с V_0 .

Всеки модул m_y сравнява собствено измерените стойности за V_{0m_y} и V_{Lm_y} използвайки хистерезис от 50mV, който компенсира неточностите при измерване и прави заключения по всяко от следните:

- **Същият или различен хранващ блок.** Ако V_{0m_y} и V_{Lm_y} измерени от модул m_y са равни това означава, че хранващият блок на m_y е различен от този на m_x , защото би трябвало да се получи поне малък пад на напрежение от натоварването на m_x , който да е видим за m_y .

- **По-далече от хранващия блок от m_x** Ако V_{0m_y} е същото, като V_{0m_x} и V_{Lm_y} е същото, като V_{Lm_x} то модулът m_y се намира по-далече от хранването, в сравнение с m_x . Важно е да се отбележи, че тъй като V_{Lm_x} е винаги по-ниско от V_{0m_x} то е невъзможно V_{Lm_y} и V_{0m_y} да бъдат с еднакви стойности. Това отличава този случай от предишният и означава, че няма смисъл да се прави извод в този

случай, ако в предишния е установено, че m_x и m_y са запазени от различни блокове. Модул m_y изчислява съпротивлението между точката на разклонение S_{xy} и запазващия блок включвайки вътрешното съпротивление на запазващия блок по (11):

$$R = \frac{V_0 - V_L}{I_L} \quad (11)$$

Тук $V_{0m_x} = V_{0m_y} = V_0$ и $V_{Lm_x} = V_{Lm_y} = V_L$. Модул m_y създава уникален номер за означаване на тази точка на разклонение, като използва уникално число съставено от последните 3 байта от собствения и MAC адреса на модул m_x , асоциира изчисленото съпротивление с този номер и ги записва в списък.

• **По-близо до запазващия блок от m_x .** Ако V_{0m_y} е равно на V_{0m_x} и V_{Lm_y} е по-голямо от V_{Lm_x} модул m_y се намира между запазващият блок и модул m_x . В този случай модул m_y може да изчисли съпротивлението между него и запазващия блок включвайки вътрешното съпротивление на запазващия блок R_1 по (12).

$$R_1 = \frac{V_0 - V_{Lm_y}}{I_L} \quad (12)$$

Както и съпротивлението между възела, от който е разклонено неговото запазване s_{yx} и модул m_x съответно R_2 по (13).

$$R_2 = \frac{V_0 - V_{Lm_y} + V_{Lm_x}}{I_L} \quad (13)$$

Тук $V_{0m_x} = V_{0m_y} = V_0$. Аналогично на предишния случай се изчислява уникален идентификатор за точката на разклонение и резултатите от изчисленията се добавят в списъка.

Всеки модул m_x направил тест с натоварване на запазващата линия също извършва изчисление на общото съпротивление между него и запазващия блок включвайки вътрешното му съпротивление по (11).

Пример: Нека приемем за момент, че m_x е m_5 и тогава според Фиг. 27 следва че:

$$V_0(m_3) = V_0(m_4) = V_0(m_5) = V_0(m_6) = V_0(m_7) = V_0(m_8) = V_0(m_9) = V_{P1}$$

$$V_L(m_3) = V_L(m_4) = V_L(m_5) = V_L(m_6) - I_L R_8 = V_L(m_7) - I_L R_8 = V_L(m_8) - I_L(R_8 + R_{12}) = V_L(m_9) - I_L(R_8 + R_{12}) = V_{P1} - I_L(R_8 + R_{12} + R_{15})$$

$$V_0(m_1) = V_0(m_2) = V_L(m_1) = V_L(m_2) = V_{P2}$$

Описаният алгоритъм не може да различи съпротивлението на кабела (линията) между първата точка на разклонение и вътрешното съпротивление на захранващия блок. Те ще бъдат разглеждани като една стойност, която всъщност е сумата от двете съпротивления и няма да бъде изрично упоменавано повече вътрешното съпротивление на захранващия блок.

Модулите изчислява и събира информация по описания начин само ако са намерили, че са свързани към същия захранващ блок с модула, който извършва натоварването. Освен уникалният номер за точките на разклонение, който беше описан те записват и те запазват към записа и идентификатор на другата точка, до която е измерено съпротивлението и идентификатор на модула извършил натоварването съставени по същия начин описан по-рано. Списъка се наименува с последните 3 байта на модула с най-малък по стойност MAC адрес, с който са свързани към общ захранващ блок.

Когато и последния модул от списъка извърши своя тест, започва обмяна на натрупаната информация между модулите, като всеки модул използва само получените списъци със същото наименование като на неговия.

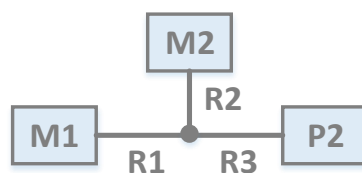
За простота нека разгледаме системата показана на Фиг. 28. В този случай ще има два списъка с еднакво име:

Съобщен от m_2 :

$$m_2, 0 = R_2 + R_3; m_1, s_{21} = R_1; s_{21}, 0 = R_3$$

Съобщен от m_1 :

$$m_1, 0 = R_1 + R_3; m_2, s_{12} = R_2; s_{21}, 0 = R_3$$



Фиг. 28 Прост пример за разпределяне на захранването

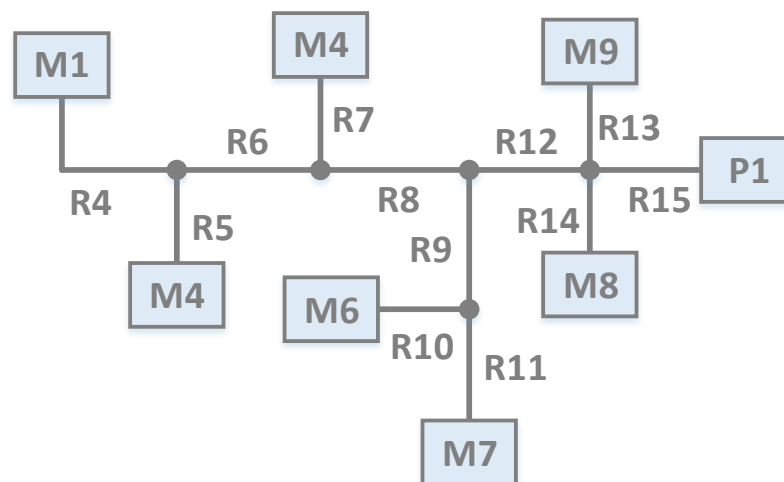
Точките на разклонение s_{21} и s_{12} са една и съща точка от перспективата на m_2 , когато теста е бил извършван от m_1 и за m_1 , когато теста е бил извършван от m_2 съответно. Модулите извършват следната процедура за кондензиране и обогатяване на информацията:

- **Обогатяване.** За всички получени записи модулет проверява дали има пълен път със съпротивления между него и захранващия блок. Ако това е вярно означава, че модулет изпратил този запис е

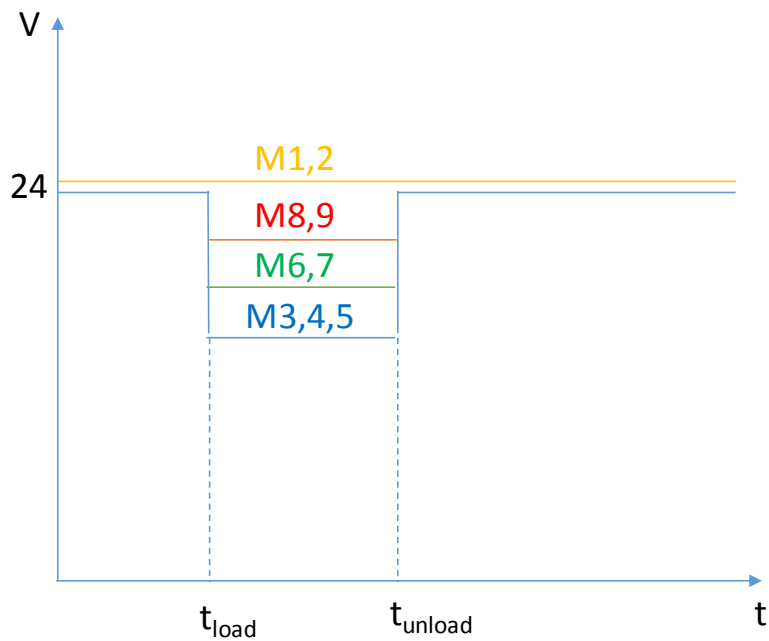
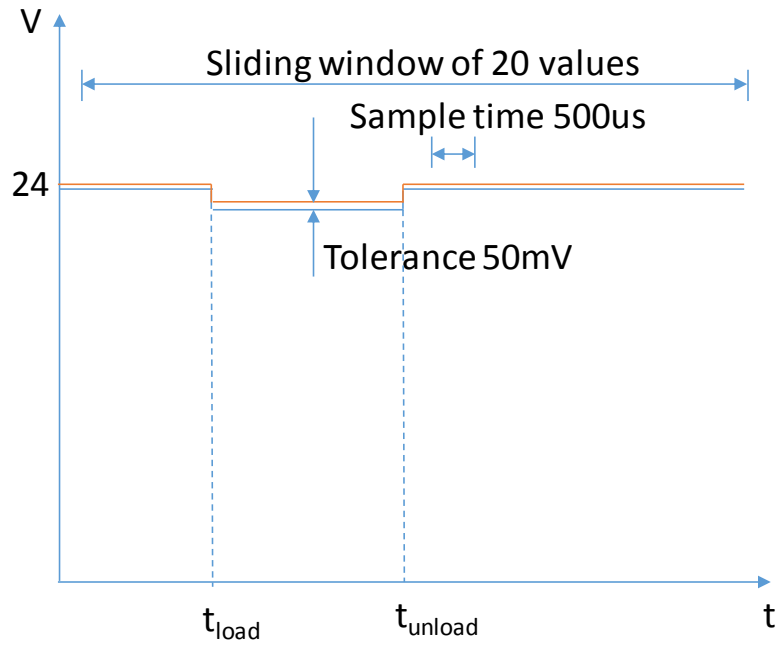
свързан някъде в през този път. Така модульт, който изследва тази информация обогатява записите, които има до момента. Например това означава, че записът $m_1, 0 = R_1 + R$ ще бъде премахнат в полза на добавянето на запис $m_1, s_{21} = R_1$ и $s_{21}, 0 = R_3$, които дават по-богата информация. Ако съществуват няколко точки на разклонение те се подреждат по отдалеченост, за която служи частично изчисленото съпротивление между модула и точката на разклонение.

- **Замяна на имената на точките на разклонение.** За всички получени записи, ако модульт намери, че е свързан някъде в участъка на пътя между модула изпратил записа и захранващия блок или обратното, че модульт изпратил записа е свързан някъде по пътя между модула получил записа и захранващия блок се прави сравнение на стойността на съпротивлението на участъка между захранващият източник и точката на разклонение като се използва малък хистерезис толериращ съвпадение [34]. При намиране на съвпадение името на точката на разклонение се замества. Например, при получена информация от m_1 за m_2, s_{12} and $s_{12}, 0$ участъци модул m_2 проверява стойността на съпротивлението от захранващия блок до първата точка на разклонение $s_{21}, 0 = R_3$ и $s_{12}, 0 = R_3$ и тъй, като те съвпадат името на точката на разклонение се замества от m_{21} на m_{12} . Използва се правило според което по-малкия номер в нотацията да се използва за сметка на по-големия, за да има уеднаквяване на имената при всички модули.

По-пълен пример на списък от втората система показана на Фиг. 27 е извадена отделно на Фиг. 29:



Фиг. 29 Реалистичен пример за разпределяне на захранването



Получено от модул m_3 :

$$m_3, 0 = R_4 + R_6 + R_8 + R_{12} + R_{15}; m_4, s_{34} = R_5; s_{34}, 0 = R_6 + R_8 + R_{12} + R_{15}; m_5, s_{35} = R_7; s_{35}, 0 = R_8 + R_{12} + R_{15}; m_6, s_{36} = R_9 + R_{10}; s_{36}, 0 = R_{12} + R_{15}; m_7, s_{37} = R_9 + R_{11}; s_{37}, 0 = R_{12} + R_{15}; m_8, s_{38} = R_{14}; s_{38}, 0 = R_{15}; m_9, s_{39} = R_{13}; s_{39}, 0 = R_{15}$$

Получено от модул m_4 :

$$m_4, 0 = R_5 + R_6 + R_8 + R_{12} + R_{15}; m_3, s_{43} = R_4; s_{43}, 0 = R_6 + R_8 + R_{12} + R_{15}; m_5, s_{45} = R_7; s_{45}, 0 = R_8 + R_{12} + R_{15}; m_6, s_{46} = R_9 + R_{10}; s_{46}, 0 = R_{12} + R_{15}; m_7, s_{47} = R_9 + R_{11}; s_{47}, 0 = R_{12} + R_{15}; m_8, s_{48} = R_{14}; s_{48}, 0 = R_{15}; m_9, s_{49} = R_{13}; s_{49}, 0 = R_{15}$$

Получено от модул m_5 :

$$m_5, 0 = R_7 + R_8 + R_{12} + R_{15}; m_3, s_{53} = R_4 + R_6; s_{53}, 0 = R_8 + R_{12} + R_{15}; m_4, s_{54} = R_5 + R_6; s_{54}, 0 = R_8 + R_{12} + R_{15}; m_6, s_{56} = R_9 + R_{10}; s_{56}, 0 = R_{12} + R_{15}; m_7, s_{57} = R_9 + R_{11}; s_{57}, 0 = R_{12} + R_{15}; m_8, s_{58} = R_{14}; s_{58}, 0 = R_{15}; m_9, s_{59} = R_{13}; s_{59}, 0 = R_{15}$$

Получено от модул m_6 :

$$m_6, 0 = R_9 + R_{10} + R_{12} + R_{15}; m_3, s_{63} = R_4 + R_6 + R_8; s_{63}, 0 = R_{12} + R_{15}; m_4, s_{64} = R_5 + R_6 + R_8; s_{64}, 0 = R_{12} + R_{15}; m_5, s_{65} = R_7 + R_8; s_{65}, 0 = R_{12} + R_{15}; m_7, s_{67} = R_{11}; s_{67}, 0 = R_9 + R_{12} + R_{15}; m_8, s_{68} = R_{14}; s_{68}, 0 = R_{15}; m_9, s_{69} = R_{13}; s_{69}, 0 = R_{15}$$

Получено от модул m_7 :

$$m_7, 0 = R_9 + R_{11} + R_{12} + R_{15}; m_3, s_{73} = R_4 + R_6 + R_8; s_{73}, 0 = R_{12} + R_{15}; m_4, s_{74} = R_5 + R_6 + R_8; s_{74}, 0 = R_{12} + R_{15}; m_5, s_{75} = R_7 + R_8; s_{75}, 0 = R_{12} + R_{15}; m_6, s_{76} = R_{10}; s_{76}, 0 = R_9 + R_{12} + R_{15}; m_8, s_{78} = R_{14}; s_{78}, 0 = R_{15}; m_9, s_{79} = R_{13}; s_{79}, 0 = R_{15}$$

Получено от модул m_8 :

$$m_8, 0 = R_{14} + R_{15}; m_3, s_{83} = R_4 + R_6 + R_8 + R_{12}; s_{83}, 0 = R_{15}; m_4, s_{84} = R_5 + R_6 + R_8 + R_{12}; s_{84}, 0 = R_{15}; m_5, s_{85} = R_7 + R_8 + R_{12}; s_{85}, 0 = R_{15}; m_6, s_{86} = R_9 + R_{10} + R_{12}; s_{86}, 0 = R_{15}; m_7, s_{87} = R_9 + R_{11} + R_{12}; s_{87}, 0 = R_{15}; m_9, s_{89} = R_{13}; s_{89}, 0 = R_{15}$$

Получено от модул m_9 :

$$m_9, 0 = R_{13} + R_{15}; m_3, s_{93} = R_4 + R_6 + R_8 + R_{12}; s_{93}, 0 = R_{15}; m_4, s_{94} = R_5 + R_6 + R_8 + R_{12}; s_{94}, 0 = R_{15}; m_5, s_{95} = R_7 + R_8 + R_{12}; s_{95}, 0 = R_{15}; m_6, s_{96} = R_9 + R_{10} + R_{12}; s_{96}, 0 = R_{15}; m_7, s_{97} = R_9 + R_{11} + R_{12}; s_{97}, 0 = R_{15}; m_8, s_{98} = R_{14}; s_{98}, 0 = R_{15}$$

Прилагайки правилата описани по-горе се правят следните опростявания:

Тъй като $m_3, 0 = R_4 + R_6 + R_8 + R_{12} + R_{15}$ изглежда, че съдържа целият път, който е получен от:

$$m_4: m_3, s_{43} = R_4; s_{43}, 0 = R_6 + R_8 + R_{12} + R_{15};$$

$$m_5: m_3, s_{53} = R_4 + R_6; s_{53}, 0 = R_8 + R_{12} + R_{15};$$

$$m_6: m_3, s_{63} = R_4 + R_6 + R_8; s_{63}, 0 = R_{12} + R_{15};$$

$$m_8: m_3, s_{83} = R_4 + R_6 + R_8 + R_{12}; s_{83}, 0 = R_{15}.$$

Може да се изведе че участъците от пътя са: $m_3, s_{43} = R_4; s_{43}, s_{53} = R_6; s_{53}, s_{63} = R_8; s_{63}, s_{83} = R_{12}; s_{83}, 0 = R_{15}$.

Използвайки второто правило и записите получени от:

$$m_3: s_{34}, 0 = R_6 + R_8 + R_{12} + R_{15};$$

$$m_4: s_{43}, 0 = R_6 + R_8 + R_{12} + R_{15}$$

Точката на разклонение с име s_{43} се замества s_{34} и използвайки записите получени от:

$$m_6: s_{63}, 0 = R_{12} + R_{15};$$

$$m_7: s_{73}, 0 = R_{12} + R_{15};$$

$$m_3: s_{36}, 0 = R_{12} + R_{15}; s_{37}, 0 = R_{12} + R_{15}$$

Точките на разклонение с имена s_{63} , s_{73} и s_{37} могат да бъдат заместени с s_{36} .

След прилагане на всички правила се достига до карта, която е описана така:

$$m_3, s_{34} = R_4;$$

$$m_4, s_{34} = R_5;$$

$$s_{34}, s_{35} = R_6;$$

$$m_5, s_{35} = R_7;$$

$$s_{35}, s_{36} = R_8;$$

$$m_6, s_{67} = R_{10};$$

$$m_7, s_{67} = R_{11};$$

$$s_{36}, s_{67} = R_9;$$

$$s_{36}, s_{38} = R_{12};$$

$$m_8, s_{38} = R_{14};$$

$$m_9, s_{38} = R_{13};$$

$$s_{38}, 0 = R_{15}$$

II.3 Методи за подобряване на комуникацията между мотора и контролера

За правилно управление на мотора, контролерът се нуждае от информация за позиция на ротора. Тя се предава по същия кабел, в който са и силовите проводници, което създава големи смущения и възможност за грешки в получената информация.

При получаване на грешна информация за позицията на ротора и извършване на грешна комутация, тока през намотките може да нарасне значително, но това е за кратък период от време, докато контролера получи правилната информация за текущото положение или следващото. Това води влошаване качеството на придвижване.

Предложени са два вида методи за подобряване на комуникацията.

- Метод за кодиране на данните чрез равно отдалечени по Hamming комбинации.
- Метод за предсказване на най-вероятното следващо състояние. Състоянието се предава 4 пъти по 3 бита. В зависимост от очакваното състояние по посока на въртенето и наличието на това състояние в поне 2 места, то го приемаме за вярно, защото е най-вероятно, според Бръсначът на Окам. [35].

II.4 Защита на устройствата от клониране

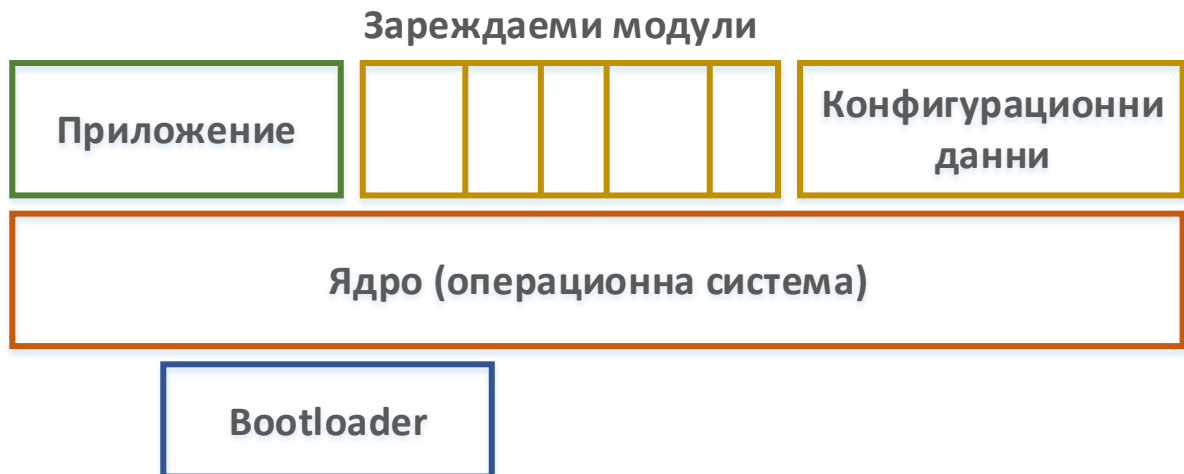
В тази част се предлагат методи и техники за защита на устройствата от копиране и съответно начини за дизайн на устройства с микроконтролери по такъв начин, който прави копирането на софтуера почти невъзможно или поне неоправдано икономически.

II.4.1 Предизвикателства при копирането на модулите за управление

Обикновено контролерът е съставен от пластмасова кутия с различни отвори, конектори, дисплей, светодиоди, стикери с надписи, отвори за закрепване, скоби и т.н. Тези неща се копират сравнително лесно, защото се виждат, могат да бъдат измерени и начертани.

В кутията обикновено са намират една или няколко печатни платки от стъклотекстолит, полиамид, алуминий, гъвкави и т.н. На тях има един или повече слоеве с медни писти и компоненти на едната или и на двете страни, като в по-новите дизайни могат да бъдат намерени компоненти и в средата на платката. Най-външните два слоя са най-лесни да копиране, защото са видими, след разпояване на компонентите, ако е необходимо. За изчертаване на вътрешните слоеве трябва да се отнемат външните слоеве медни писти и изолация (т.н. препрег) и така докато бъдат пречертани всички слоеве. Ако компонентите не са брандирани или направени по поръчка те обикновено могат да бъдат разпознати и да бъдат поръчани от съответните доставчици.

В съвременните устройства съдържащи микроконтролер(и) софтуера може да бъде разделен на няколко части Фиг. 30:



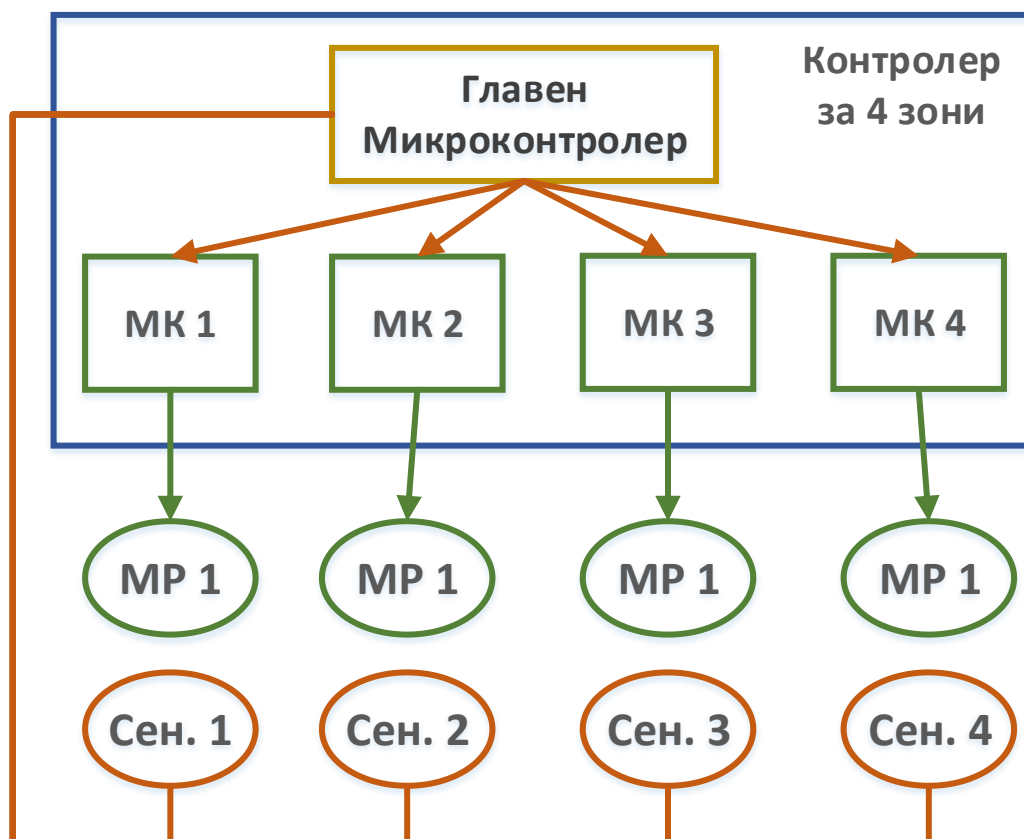
Фиг. 30 Видове софтуер в постоянната памет на контролера

- **Bootloader** – Записана веднъж програма и рядко с възможност да бъде презаписвана след време. Служи за подготовка и стартиране на основния софтуер.
- **Ядро** – Това е самата операционна система, ако устройството работи с такава или скелета с базови функции на основния софтуер.
- **Приложение** – Софтуера изпълняващ специфичните функции на контролера, където са алгоритмите и основното know-how.
- **Зареждаеми модули** – Понякога, когато паметта на устройството не е достатъчна за всички възможни функционалности, които модула може да изпълнява те се разделят на модули и се записва само модула за конкретното приложение.
- **Памет за постоянни данни** – Различни калибрационни коефициенти, пароли, запазени стойности, параметри и т.н.

Понякога няма необходимост от сменяне на която и да е част от софтуера, за периода на целия живот на изделието – например този в мотора, който кодира серийно информацията за датчиците за позиция на ротора. Тогава могат да бъдат използвани т.н. OTP ROM памети, където след като бъде записано веднъж нещо то не може да бъде изтривано и променяно.

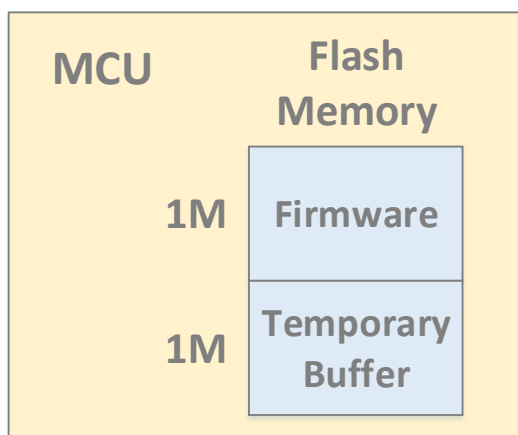
В случаите, когато приложния софтуер трябва да се подмени, заради обновяване на сигурността, добавяне на нова функционалност или промяна на съществуваща такава обикновено се използва Flash памет, а в случаите когато броя презаписвания е сравнително голям – напр. стойности на параметри и коефициенти се използва EEPROM или RAM с допълнително захранване от батерия. Не винаги тези видове памети и/или необходимите

размери са налични в един микроконтролер, което налага използването на отделни чипове за това и на външни шини за комуникация.



Фиг. 31 Архитектура на контролер управляващ четири зони

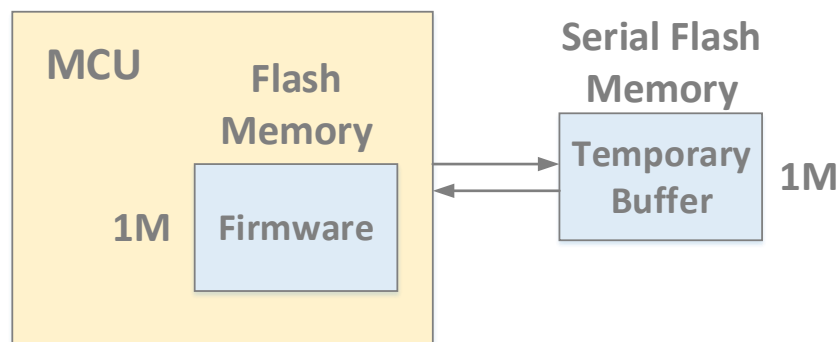
Друг случай, в който се налага използването на шини за комуникация е когато не може да се реализира цялото управление в един микроконтролер – напр. контролер за конвейер управляващ 4 зони Фиг. 31. Въртенето на 4 БЧПТ двигателя изисква 12 ШИМ (Широчинно Импулсно Модулирани) изхода, което рядко се намира в един контролер.



Фиг. 32 Използване на половината вътрешна памет за временно място по време на обновяване на софтуера

Външните шини могат да бъдат подслушвани и ако данните по тях не са криптирани те могат да бъдат записани синхронно заедно с поведението на контролера, и така да бъде разгадано тяхното значение [36]. И докато разгадаването на значението на данните може да не е чак толкова голяма беда, по някои шини може да извършва обновяването на софтуера, което при правилно подслушване дава самия софтуер на устройството.

Да се избегнат външните шини не е чак толкова лесно, поради това, че по време на обновяване на софтуера, трябва да се ползва временна памет, където да се записват данните идващи от комуникационния канал и след като бъде проверена целостта и автентичността им със съответните суми за проверка да бъдат записани на мястото от където ще се изпълнява кода Фиг. 32. Това на практика намаля използваемата памет наполовина и понякога води до това, че от съответната фамилия контролери няма такъв с достатъчно памет. Тогава се налага използването на външна флаш памет Фиг. 33.



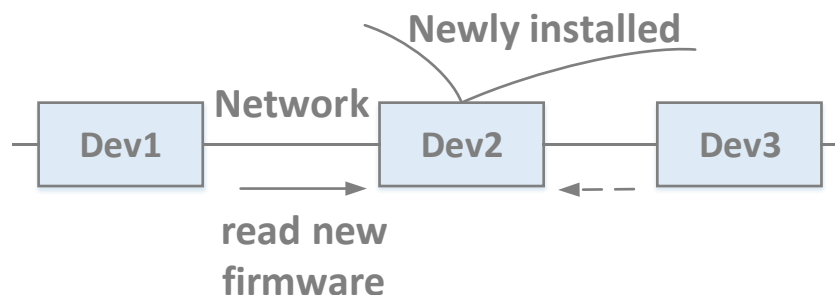
Фиг. 33 Използване на външна памет за временно място при обновяване на софтуера

Друга причина, която налага използването на външна памет е много по-ниската ѝ цена в сравнение на микроконтролер от същата фамилия и същото количество по-голяма памет.

Друг подход за придобиване на изпълнимия код на микроконтролерите в устройството е чрез подслушване на главния комуникационен канал, който контролера използва за връзка с останалите контролери и софтуера за управление и конфигуриране. Такива канали могат да са Ethernet, USB, Bluetooth, Wi-Fi, Zigbee и т.н.

Много често в контролерите за конвейери произвеждани от един производител има заложили функции на само-възстановяване и само-обновяване на софтуера на ново-свързания контролер от останалите в системата. Механизмът извършва сравнение на версиите на софтуера от съседните модули и своята и в случай, че е с по-стара извършва автоматично

обновяване, като прочита новата версия от някой от съседните модули Фиг. 34.



Фиг. 34 Само-обновяване на софтуера от съседните модули

Този механизъм предотвратява нежеланото появяване на стари проблеми свързани със софтуера, които са били поправени докато системата е била експлоатирана с периодични обновявания на софтуера и когато дефектирал контролер бъде сменен с такъв от склада, който е с по-стара версия на софтуера. С каквато са били закупени всички контролери в началото.

Комуникационният канал обаче може да бъде подслушван в момента на това само-обновяване и да бъде записан целия софтуер.

Копирането се обезсмисля, когато вложените средства за копиране надхвърлят средствата необходими за проектиране на ново устройство. Затова фокуса на предложените тук методи е към използването на техники затрудняващи копирането

Всички видими части на устройството, като кутия, разположение на компонентите и т.н. е невъзможно да бъдат скрити и затова е лесно да бъдат копирани Фиг. 14.

Обикновено имената на електронните компоненти могат да бъдат изтрети с абразивен инструмент или разтворител, в зависимост технологията на принтиране. Това обаче изисква допълнителни усилия и увеличава себестойността на устройството затова рядко се използва. Ако обема на произвежданите изделия е голям могат да бъдат поръчани компоненти от производителя със специални наименования, за които да не може да се намери информация в интернет.

Един от начините за откриване на името на електронен компонент е да се познае неговата функция и на база името на корпуса и стойностите на съседните елементи да се търси примерен дизайн в Интернет. Трябва да се отбележи, че корпусите на елементите са стандартизирани и опитните инженери по електроника ги познават на външен вид. Този начина на обратно инженерство [20] обикновено дава добри резултати, защото

производителите на електронни устройства следват примерните дизайни на производителите на електронни компоненти.

Добър начин за затрудняване на копирането на печатната платка е скриването на колкото може повече сигнални писти във вътрешните слоеве на платката [37]. Начин за разкриване на тези връзки е след, като се разпоят всички компоненти да бъдат отнемани постепенно най-външните слоеве на платката с абразивна хартия, докато се покаже следващия слой. Обикновено това е сравнително трудна задача, защото вътрешните медни слоеве са сравнително тънки 35 микрона и е сравнително лесно да бъдат премахнати по грешка.

Проходният отвор в печатната платка служи за свързване на пистите в различните слоеве. Обикновено те преминават през всички слоеве, независимо кои свързват, защото това улеснява производството на платката. Съществуват обаче и т.н. глухи отвори, които повишават стойността на платката, но дават възможност за по-висока гъстота на компонентите. В този случай копирането е по-трудно защото не са достъпни за измерване на свързаност с омметър през външните слоеве [38].

Методът за измерване на свързаност през проходните отвори може да отнеме изключително дълго време при по-големи печатни платки, защото всички комбинации трябва да бъдат измерени. Електронен инженер с по-голям опит може да отгатне ориентировъчно местата, които трябва да са свързани по функцията на компонентите и да намали това време значително.

II.5 Добавяне на интелигентност чрез използване на невронна мрежа

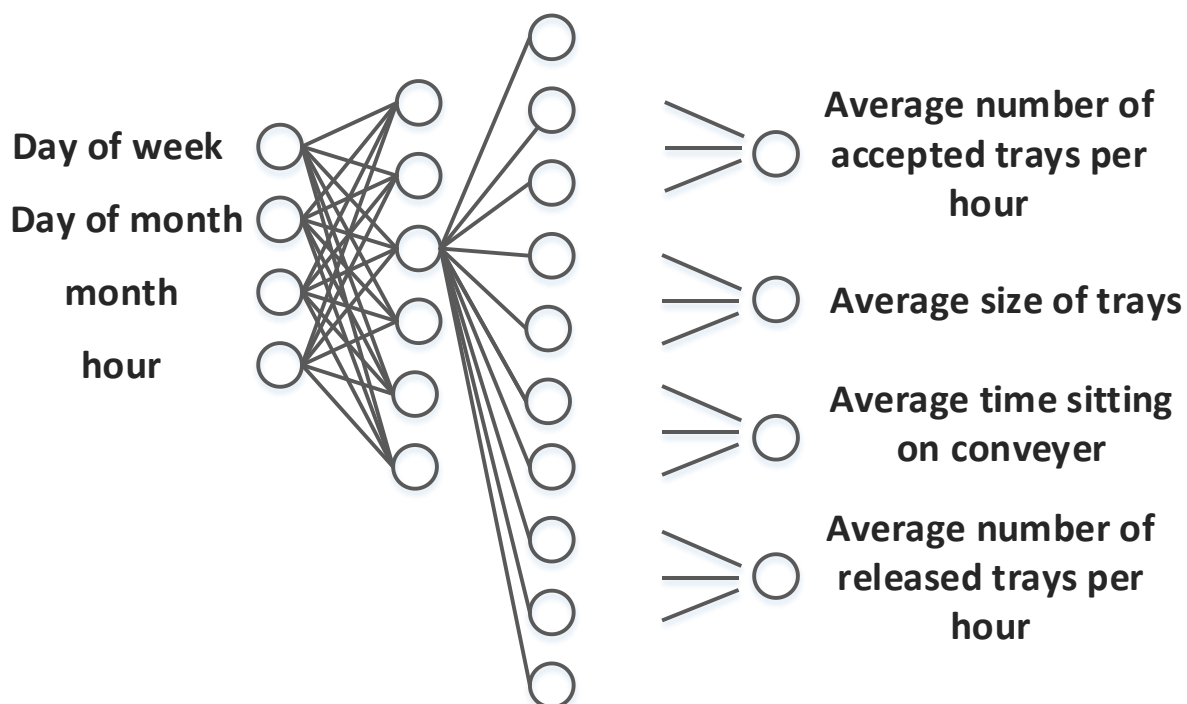
Основната цел е да се разработи метод за динамично подобряване характеристиките на конвейерните системи, който чрез използване разпределена невронна мрежа намират зависимости в натоварването и позволяват на конвейера да се адаптира превантивно към промените. С други думи, конвейера ще знае, кога да се опитва да акумулира повече товари и кога да се опитва да ги придвижи максимално бързо.

За целта се анализират различни методи за моделиране на натовареността на системата спрямо времето като например *mixture models*.

Направените до момента проучвания показват, че проблемът се решава добре с напълно свързана Feed-Forward невронна мрежа Фиг. 35.

За входове на мрежата са избрани ден от месец, ден от седмица, час

Като изходи са дефинирани средното количество пратки влизаци и излизаци, средна скорост на придвижване на пратките.



Фиг. 35 Структура на Feed-forward невронна мрежа

Ако се открие аномалия, то трябва да се определи периодът на повторяемост.

При използване на онлайн обучение, се очаква тежестите съответстващи на най-бързо променящите се входове ще се променят на най-бързо. Например, първо ще се открият тежестите, които се тренират за 24 часа, ако съществува някаква периодичност в този интервал. След това за седмица и след това за месец.

Ще бъдат разработени два подхода за разпределено трениране на мрежата.

Всеки слой ще се разделя и разпределя върху контролерите на зоните, след което данните се обединяват и се преминава към изчисление на следващия слой.

Всяка зона ще бъде разглеждана като автономен агент, който може да изчислява тежестта на възлите само по някой от избраните изходи. След това агентите могат да обменят информация за прогнозните стойности на параметрите за които са избрани да тренират невронната мрежа.

При различно представяне на входните данни, може да се изследва възможността да се моделира цикличност чрез тригонометрични функции.

В този случай може да се наложи създаване на собствени функция за активация, защото в например в TensorFlow няма имплементирани такива.

Основният фокус пада върху намиране на подходящо представяне на входните данни, като интерес представлява намирането на корелации.

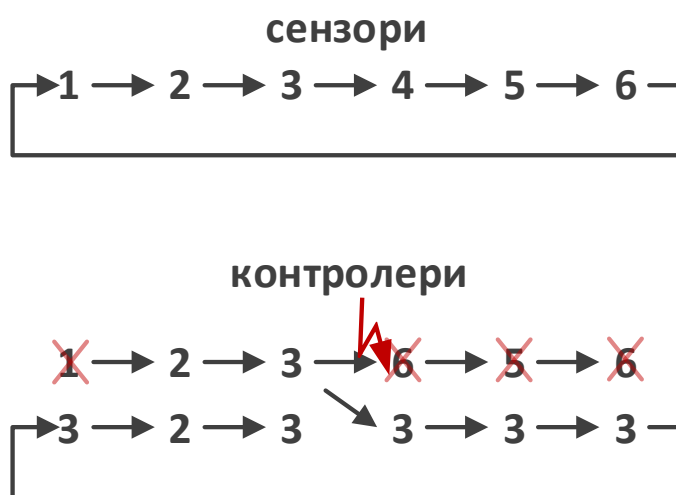
Ще се изследва също така и възможността да се използва машина Болцман за намиране на зависимост между входа/времето и натовареността.

III ГЛАВА. ИМПЛЕМЕНТАЦИЯ НА ПРЕДЛОЖЕНИТЕ РЕШЕНИЯ

III.1 Имплементация на предложеният метод за подобряване качеството на данните за позицията на моторите

За целите на определяне на глобалната позиция на ротора, където за всеки преход в ново състояние се отчита и посоката на завъртане, такива грешки могат да доведат до натрупване на обща грешка по позиция и да компрометират метода за позициониране. Затова се налага да се търсят подходи за филтриране и подходящо обработване на такива ситуации, като ако информацията не е очакваното положение в някоя от двете посоки на въртене или текущата се прави следното:

- Информацията се игнорира. Такова решение е не винаги приемливо, защото както е показано на Фиг. 36 ако бъде игнорирана информацията за следващо положение, заради зашумяване, то текущото положение няма да бъде обновено и така по-следващата позиция, която може да е валидна ще бъде също игнорирана.



Фиг. 36 Ефект от игнорирането на погрешна информация от датчиците за позиция

- Приема се за вярна следващата позиция в текуща посока на въртене. Въпреки, че този метод има недостатък само, когато се сменя посоката на въртене, резултата в тези редки случаи е много по-голямо раз-синхронизиране на последователностите, както е показано на Фиг. 37.

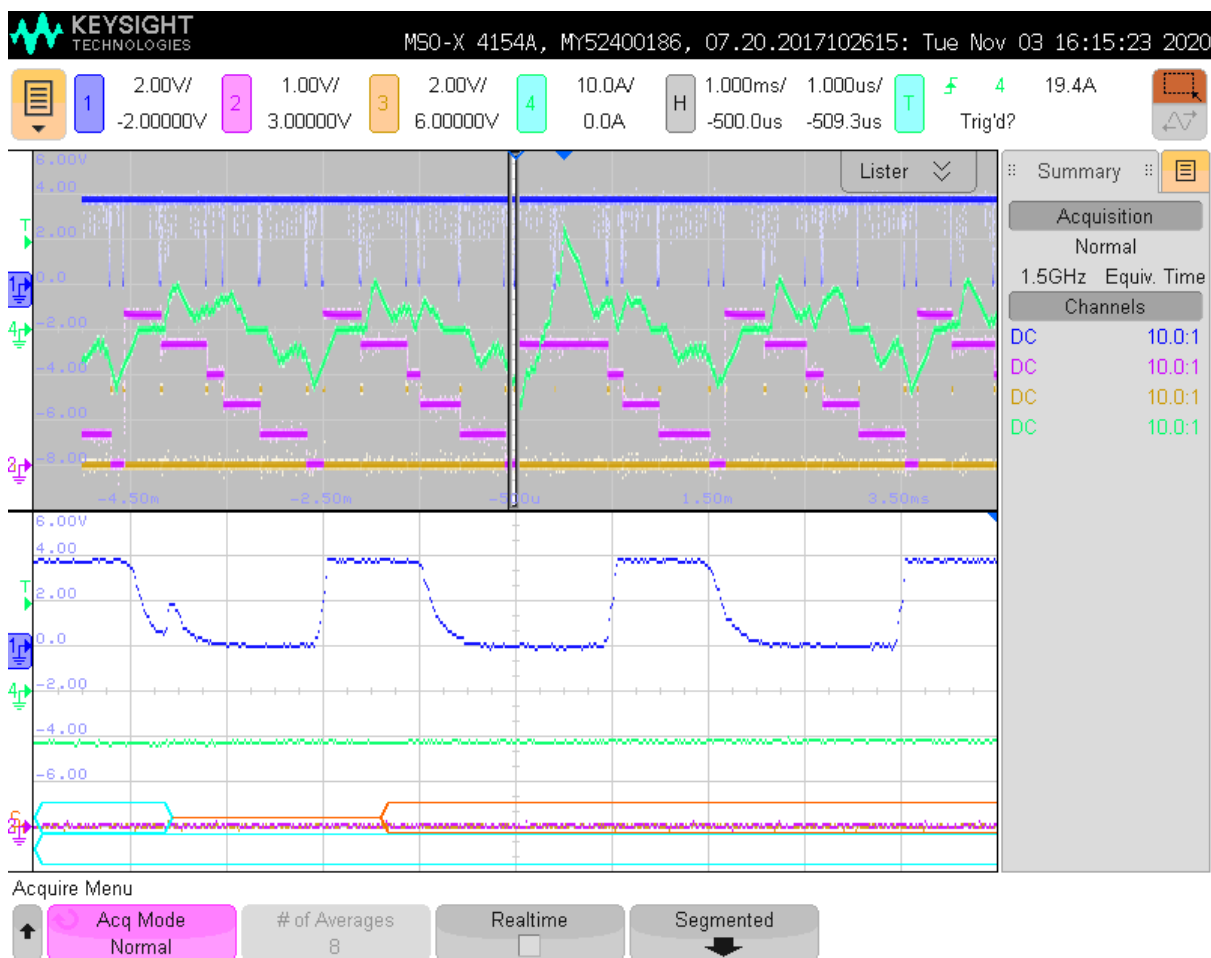


Фиг. 37 Ефект от приемането на следващата по посока на въртене информация при получаване на грешна такава.

Прилагат се и методи в които се отчита времето през които трябва да пристигат информациите за положението на ротора, като въз основа на тази стойност се формира прозорец от време в който се очаква следващата валидна информация и тя се третира, както беше описано във втория случай, а новата информация пристигнала извън този прозорец се игнорира както беше описано в първия случай.

Този метод обаче има недостатъци при рязка смяна на скоростта на въртене на ротора, което се случва при задвижването на механизми с крайни положения Фиг. 48 и Фиг. 52.

На Фиг. 38 може да се види осцилограма, показваща влиянието на получената грешна информация от датчиците за позиция на ротора, отразена с розово, върху консумирания ток от мотора отразен със зелено. В средата на осцилограмата се вижда, че стъпаловидното преместване на позицията пропуска най-горната позиция и това води до моментално повишаване на тока. С тъмно синьо е показан сигнала от комуникационния канал, като той е увеличен в долната половина на осцилограмата и там се вижда смущението индуцирано в него.



Фиг. 38 Влияние на получената грешна информация за положението на ротора върху консумирания ток от мотора

Разработен е софтуер, който да намери равно отдалечени по Хаминг [39] комбинации, които да бъдат използвани при предаването на информация, като част от кода и външния вид на приложението с изведените резултати са показани на Фиг. 39 и Фиг. 40.

Резултатът от тази програма е използван в софтуера на моторизирания ролер и контролера за управление за кодиране и декодиране на предаваната информация за положението на ротора на двигателя.

```

namespace CRC4_hamming_distance {
public partial class Form1 : Form {
public Form1() {
InitializeComponent();
List<byte> msg = new List<byte>();
for (byte val = 8; val < 0x10; val++) {
byte crc = crc4(0, val, 4);
byte dat = (byte)(0x80 | crc << 3 | val & 7);
msg.Add(dat);
lbVals.Items.Add("0x80 | crc=" + crc.ToString("X2") + "<<3 | val=" + (val & 07).ToString("X2") + ") -> " + ".");
}
lbVals.Items.Add("Differences between message bits:");
foreach (byte a in msg) {
string s = "";
foreach (byte b in msg)
s += iHammingDistance(a, b) + ", ";
lbVals.Items.Add(s);
}
lbVals.Items.Add("Array with bytes on Hamming distance 1:");
lbVals.Items.Add("Copied to clipboard starting with the original byte.");
List<List<byte>> llbHam1 = new List<List<byte>>();
string clpBrd = "";
foreach (byte a in msg) {
List<byte> lb = lbGetBytesOnHammingldist(a);
llbHam1.Add(lb);
lbVals.Items.Add(sPrintByteList(lb, false));
clpBrd += "0x" + a.ToString("X2") + ", " + sPrintByteList(lb, true) + "\\r\n";
}
Clipboard.SetText(clpBrd);
foreach (List<byte> lba in llbHam1)
foreach (byte a in lba) {
lbVals.Items.Add("Hamming distance of " + a.ToString("X2"));
foreach (List<byte> lbb in llbHam1) {
List<byte> df = new List<byte>();
bool bFoundLessThanTwo = false;
foreach (byte b in lbb) {
if (b == a)
continue;
byte diff = (byte)iHammingDistance(a, b);
if (diff < 2)
bFoundLessThanTwo = true;
df.Add(diff);
}
lbVals.Items.Add(sPrintByteList(df, false) + (bFoundLessThanTwo ? "Found distance below 2!" : ""));
}
}
}
}

byte[] crc4_tab = { 0x0, 0x7, 0xe, 0x9, 0xb, 0xc, 0x5, 0x2,
0x1, 0x6, 0xf, 0x8, 0xa, 0xd, 0x4, 0x3, };
private byte crc4(byte c, uint x, int bits) {
int i;

/* mask off anything above the top bit */
x &= ((uint)1 << bits) - 1;

/* Align to 4-bits */
bits = (bits + 3) & ~0x3;

/* Calculate crc4 over four-bit nibbles, starting at the MSbit */
for (i = bits - 4; i >= 0; i -= 4)
c = crc4_tab[c ^ ((x >> i) & 0xf)];

return c;
}

int iHammingDistance(byte b1, byte b2) {
byte diff = (byte)(b1 ^ b2);
int dist = 0;
for (int i = 0; i < 8; i++)
if ((diff & 1 << i) != 0)
dist++;
return dist;
}

List<byte> lbGetBytesOnHammingldist(byte b) {
List<byte> lb = new List<byte>();
for (int i = 0; i < 8; i++) {
lb.Add((byte)(b ^ (1 << i)));
}
return lb;
}

string sPrintByteList(List<byte> bl, bool bPrep0x) {
string s = "";
foreach (byte b in bl) {
if (bPrep0x)
s += "0x";
s += b.ToString("X2") + ", ";
}
return s;
}
}
}

```

Фиг. 39 Имплементация на софтуера генериращ комбинациите по Хаминг

```
Form1
(0x80 | crc=01<<3 | val=00) -> 88
(0x80 | crc=06<<3 | val=01) -> B1
(0x80 | crc=0F<<3 | val=02) -> FA
(0x80 | crc=08<<3 | val=03) -> C3
(0x80 | crc=0A<<3 | val=04) -> D4
(0x80 | crc=0D<<3 | val=05) -> ED
(0x80 | crc=04<<3 | val=06) -> A6
(0x80 | crc=03<<3 | val=07) -> 9F
Differences between message bits:
0, 4, 4, 4, 4, 4, 4, 4,
4, 0, 4, 4, 4, 4, 4, 4,
4, 4, 0, 4, 4, 4, 4, 4,
4, 4, 4, 0, 4, 4, 4, 4,
4, 4, 4, 4, 0, 4, 4, 4,
4, 4, 4, 4, 4, 0, 4, 4,
4, 4, 4, 4, 4, 4, 0, 4,
4, 4, 4, 4, 4, 4, 4, 0,
Array with bytes on Hamming distance 1:
Copied to clipboard starting with the original byte.
89, 8A, 8C, 80, 98, A8, C8, 08,
B0, B3, B5, B9, A1, 91, F1, 31,
FB, F8, FE, F2, EA, DA, BA, 7A,
C2, C1, C7, CB, D3, E3, 83, 43,
D5, D6, D0, DC, C4, F4, 94, 54,
EC, EF, E9, E5, FD, CD, AD, 6D,
A7, A4, A2, AE, B6, 86, E6, 26,
9E, 9D, 9B, 97, 8F, BF, DF, 1F,
Hamming distance of 89
02, 02, 02, 02, 02, 02, 02,
04, 04, 04, 02, 02, 02, 04, 04,
04, 04, 06, 06, 04, 04, 04, 06,
```

Фиг. 40 Резултат от работа на програмата намиращата равно отдалечени комбинации по Хаминг

III.2 Имплементация предложения метод за подобряване на захранването

За всеки модул, свързан към захранване, са зададени две конфигурационни стойности. Първата е максималният ток, който захранването може да осигури, без да се включи защитата от претоварване а втората е максималното напрежение, което може да толерира на изходните си терминали, без да се включи защитата от пренапрежение. Модулът изпраща актуализира информация за тези стойности на всички останалите модули в групата.

Всеки модул отчита за общия ток, консумиран от захранването. Той актуализира тази стойност от съобщенията, които останалите модули в групата изпращат. Модулът е длъжен да изпрати актуализация, ако

текущото му потребление се промени извън предварително конфигуриран праг, първоначално зададен на 0,5А.

Message data		
Total current	Module current	Incremental current

Фиг. 41 Формат на съобщението за консумиран ток

В съобщението показано на Фиг. 41 се съдържат три поета: за общата стойност I_T , действителният текущ модул I_M и инкременталната част I_I която задейства актуализацията. Модулът получава общата стойност и актуализира вътрешната си стойност само ако е изтекъл предварително зададен лимит преди и след приемането на съобщението, който първоначално е зададен на 100ms. В противен случай той използва само нарастващата стойност I_I за да актуализира вътрешната си стойност. Този механизъм гарантира, че ако две съобщения са изпратени с много кратко закъснение между тях, получателят няма да обърка стойността на общия ток. Това може да се случи, ако последният изпращащ модул не е получил и декодирал предишното съобщение и изпрати общ ток, без да включва нарастването на предишния модул.

Всички модули поддържат списък с консумацията на останалите модули в групата, за да проверят общия ток през предварително определени редовни интервали, които първоначално са зададени на 10sec. Модулите изпращат съобщение за актуализация с техните токове I_M and I_I през този интервал.

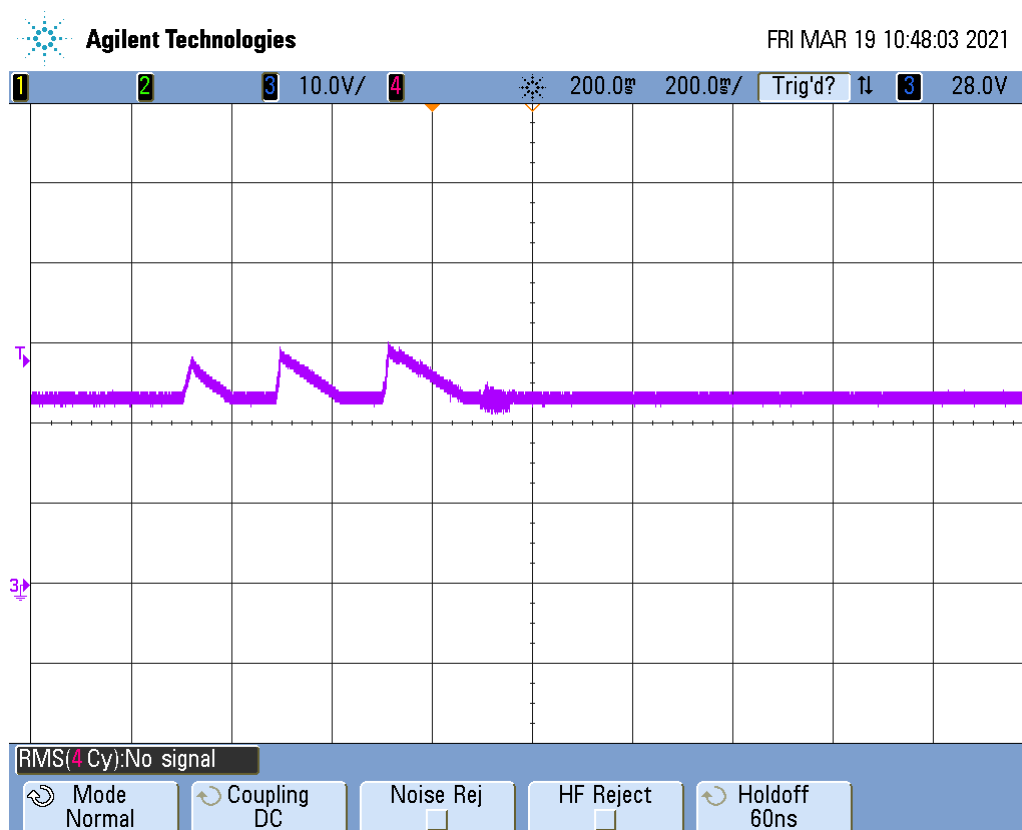
Първият основен момент на алгоритъма е, че ако модулът трябва да консумира ток, той проверява общия ток, консумиран в момента, и общия допустим ток от захранването и ако токът е недостатъчен, той използва само наличната част от него. Това гарантира, че захранването няма да изключи настоящата си защита.

Вторият основен момент е, че когато модул се опитва да забави скоростта на своя двигател и генерира енергия, той може да върне към електропровода частта, която е равна на общия консумиран в момента ток I_T като се извади някакъв предварително дефинирана марж, първоначално зададена на 0,5А. Тази разлика е необходима, защото ако общият ток е близо до 0, той може да стане отрицателен за кратък период от време. Ако това се случи, то напрежението на електропровода може да се повиши и защитата от пренапрежение на захранването може да сработи.

Модулът, който връща енергията, изчислява на спада на напрежението, започвайки от захранването, докато включва максимално допустимото предварително зададено напрежение.

Продължавайки надолу по линията и използвайки съпротивления на всяка част, както и списък с токовете, които всеки модул консумира в момента, то по законите на Кирхоф и Ом, модулът изчислява максималния ток, който може да върне, ограничен от напрежението I_{MV} . Модулът избира по-ниската стойност между $I_T - 0.5$ и I_{MV} и задава I_M като отрицателната стойност на тази стойност. Стойностите I_I and I_T се изчисляват по аналогичен начин. Модулът изпраща съобщение с тези стойности и ако се окаже, че модулът трябва да върне повече ток от изчислената от него стойност, той използва своята компенсационна верига и го разсейва като топлина.

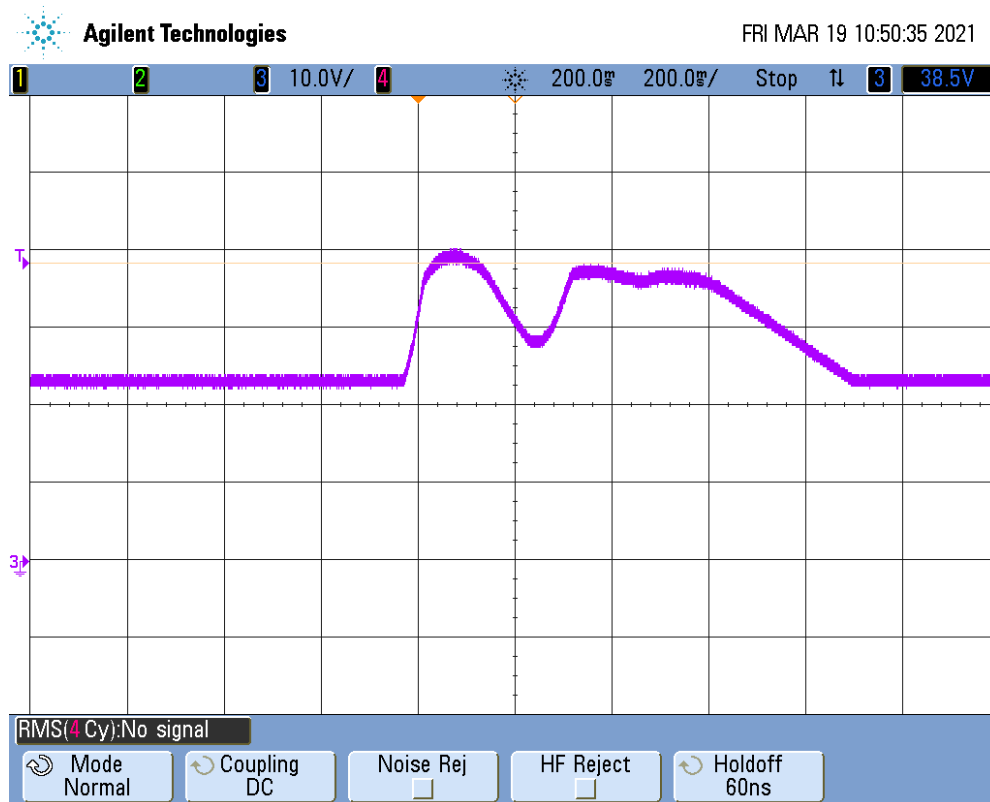
Възможността за използване на захранващи устройства, чиито сумарен максимален ток е много по нисък от номиналния ток на всички инсталирани модули за управление намалява значително цената на конвейера. В момента проектантите правят такива конвейерни инсталации на свой риск. Използването на описания алгоритъм елиминира случайното активиране на защитата и единственото въздействие е влошаването на производителността на системата за някакъв кратък период от време (в зависимост от това колко слаб захранващ блок е бил използван).



Фиг. 42 Повишаване на захранващото напрежение при спиране на малък обект.

Обикновено генерираната енергия от модулите по време на забавяне скоростта на товарите се преобразува на 100% в топлина. Чрез използване на описания алгоритъм част или дори цялата енергия може да бъде използвана повторно и по този начин да се постигне много по-добра енергийна ефективност. Колкото повече модули са свързани едно захранване, колкото повече енергия може да се възстанови, тъй като има по-голям шанс друг модул да се нуждае от енергия в момента на връщането ѝ. Освен това повече модули означават повече работен ток - който също се създава от върнатата мощност.

На Фиг. 42 и Фиг. 43 е показано, как напрежението на захранващия блок се повишава, при спиране на различни по тежест обекти. В случая на по-тежък обект, които е натрупал значително по-голяма кинетична енергия при спирането си повишението е също значително.

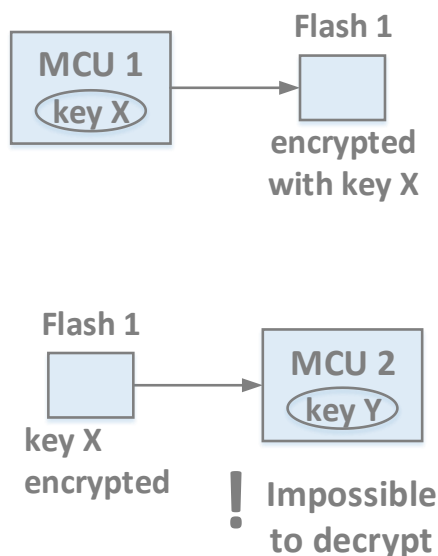


Фиг. 43 Повишаване на захранващото напрежение при спиране на тежък обект.

III.3 Имплементация на метода за защита на модулите от клониране

За да се намали възможността да бъде използван софтуер, който е придобит с подслушване на комуникационен канал той трябва да бъде криптиран. Например при използване на външна временна памет, данните биват криптирани, прехвърлени отвън и след като цялото прехвърляне

завърши те биват четени, декриптирани и записвани във вътрешната памет Фиг. 44.

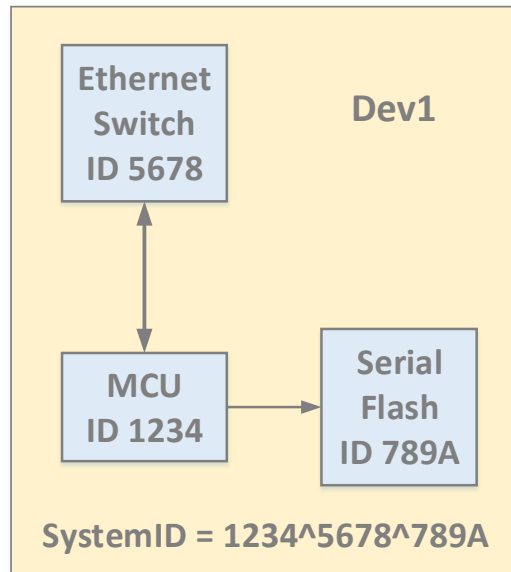


Фиг. 44 Криптиране на данните

Въпреки, че това изглежда като надежден метод против подслушване, защото подслушаните данни няма да значат нищо, той не може да предпази от копиране на цялото устройство, ако ключа за криптирането не зависи от нещо, което е различно в отделните електронни компоненти. В такъв случай устройство със същия хардуер ще декриптира успешно презаписана в нова памет подслушана информация.

В ключа трябва да участва например сериен номер на микроконтролера, който производителя на чипа записва и е винаги различен.

Както е споменато в предишната част, криптирането на фърмуера трябва да зависи от уникалния идентификатор на чипа. Такива стойности обикновено са налични в много от чиповете, чрез които се изгражда системата, например, микроконтролер, серийна флаш памет и др. Те могат да се използват за изграждането на “пръстов отпечатък” на устройството, който се използва в криптирането на фърмуера и данните, Фиг. 36. Това ще предотврати възможността устройството да работи със заменени чипове, а също така ако някой от чиповете бъде премахнат, то той да работи на друго устройство.



Фиг. 45 Изграждане на пръстов отпечатък на устройството, което се използва за криптиране на фърмуера и данните.

Обикновено микроконтролерите, дори и с деактивиран JTAG достъп, все още предоставят възможност да бъдат изтрети или чрез отделен пин, или с JTAG команда за “изтриване на чип” (тази би била единствената възможна команда).

Даден хакер може да записва комуникацията при обновяване на фърмуера, да изтрие чипа за да се изчисти конфигурацията, която деактивира JTAG достъпа, след което да прочете неговия идентификатор. Серийни номера на Ethernet ключ и серийна флаш и др. памет могат лесно да бъдат прочетени, дори ако паметта има защитени страници. В крайна сметка ключът на устройството може да бъде възстановен. За да се избегне такава възможност, се препоръчва използването на асиметрично криптиране и публичен ключ, базиран на случайни числа.

В случай, че за комуникацията с външния свят се използва симетрично криптиране, е добре към ключа да се добави някаква стойност, известна и за двете страни (IoT устройството и компютъра), като например “TCP window sequence” номера, MAC адреси и т.н. Това ще направи шифрования файл винаги различен и за хакера ще бъде много по-трудно да осъзнае какво причинява разликата.

Предложеният метод за подsigуряване на контролерите срещу клониране е представен на 9th Balkan Conference on Informatics:

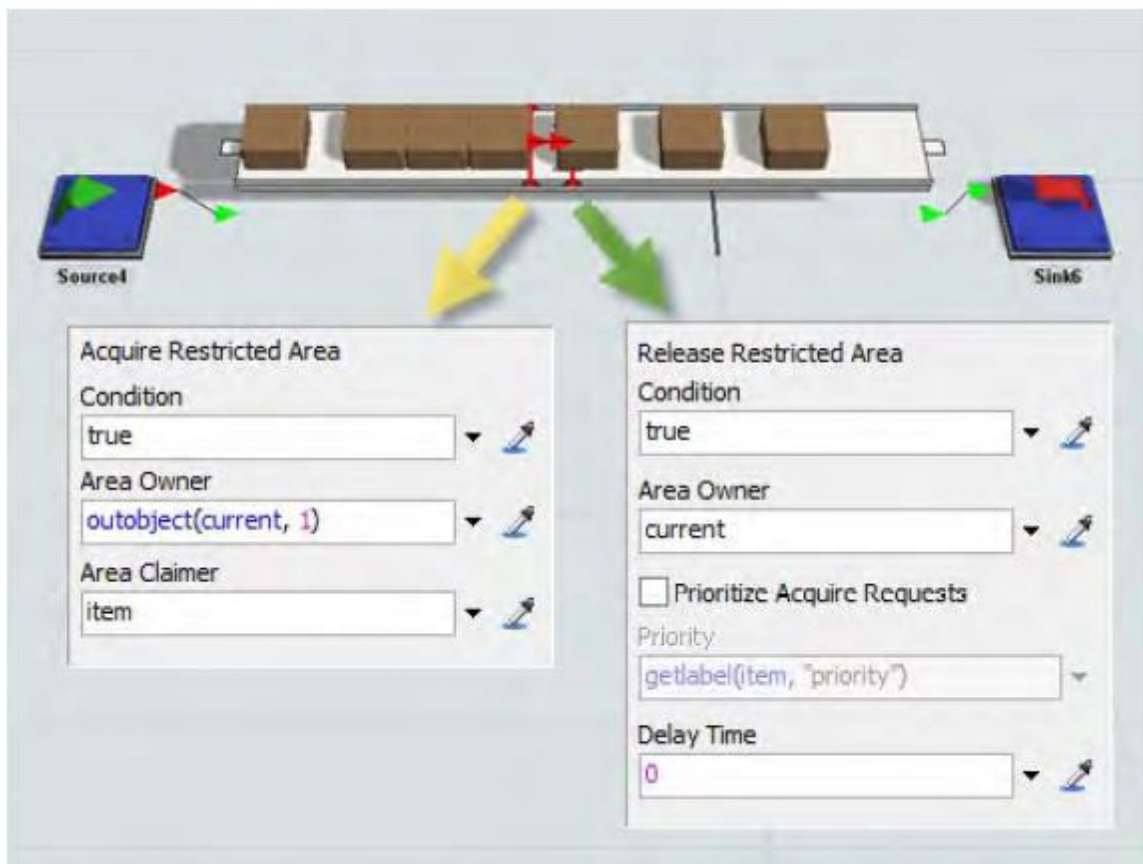
Ivailo Andonov, Simeon Tsvetanov, Stefan Dimitrov, Securing IoT devices against cloning, COMPUTER & COMMUNICATIONS ENGINEERING, vol:13, issue:2, 2019, pages:25-28, ISSN (print):1314-2291

III.4 Имплементация на предложения метод за разпределено управление чрез компютърни симулации

Софтуерът за симулации FlexSim предлага възможност за писане на скриптове, които да извършват изчисления при случването на събития на различни места на симулирания конвейер [40]. Създаден е код, който реализира идеята за динамично променяне на празното разстояние между обектите, чийто код е показан на Фиг. 46.

```
Conveyor current = ownerobject(c);
Object item = param(1);
{ // Cr2
    Object involved = item;
    //current.targetSpeed = current.currentSpeed * 2.7;
    int traysNum = current.subnodes.length;
    int i;
    double len = 0.0;
    for (i = 1; i <= traysNum; i++) { // find the total box len
        Object o = current.subnodes[i];
        len += o.size.x;
    }
    if (len > current.size.x)
        len = current.size.x - traysNum * 0.04;
    double lenUp = (current.prev.trayUp? == nullvar) ? 0.01 : current.prev.trayUp;
    double gapUp = (current.prev.gapUp? == nullvar) ? 1 : current.prev.gapUp;
    double lenDn = (current.next.trayDn? == nullvar) ? 0.01 : current.next.trayDn;
    double gapDn = (current.next.gapDn? == nullvar) ? 1 : current.next.gapDn;
    current.labels.assert("gapUp", 0).value = gapUp + current.size.x - len;
    current.labels.assert("trayUp", 0).value = lenUp + len;
    current.labels.assert("gapDn", 0).value = gapDn + current.size.x - len;
    current.labels.assert("trayDn", 0).value = lenDn + len;
    double densUp = lenUp / gapUp;
    double densDn = lenDn / gapDn;
    double desiredSpeed;
    if (densDn == 0)
        desiredSpeed = 1;
    else
        desiredSpeed = densUp / densDn;
    if (desiredSpeed > 1)
        desiredSpeed = 1;
}
```

Фиг. 46 Код за реализиране на динамично променяне на празното разстояние между обектите



Фиг. 47 Задаване на параметри на вградената функционалност за създаване на празно разстояние между обектите

Софтуерът има някои вградени функционалности, които могат да бъдат използвани само с настройването на параметри в някои менюта Фиг. 47.

III.5 Имплементация на предложения метод за разпределено управление върху реална система

III.5.1 Актуатори за конвейерни системи

Освен споменатите вече моторизирани ролери за задвижване ролките на конвейера съществуват и други актуатори, предизвикващи придвижване на обектите в напречна или вертикална посока. Общото между всички актуатори в повечето случаи е, че се задвижват от безчеткови постояннотокови двигатели. В по-редки случаи се използва пневматика, защото тя изисква въздухо-подготвящи групи, чиято себестойност не е оправдана, ако останалата част от системата не използва пневматика.



Фиг. 48 Напречно избутване



Фиг. 49 Изскачащи колелца

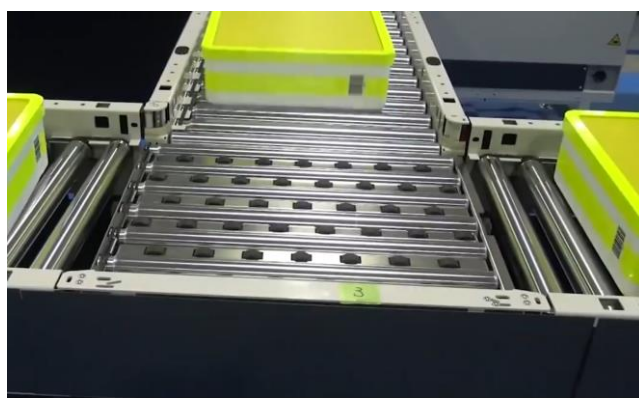
Устройствата [41] за напречно придвижване са чрез избутване Фиг. 48, с изскачащи колелца Фиг. 49, завъртаща се маса [42] Фиг. 50, както и с повдигащи се ролки [43] Фиг. 51 [44] или ремъци [7] Фиг. 52, могат да променят ориентацията на обекта по посоката му на движение или да я запазят.

Промяната на ориентацията на обекта в посоката му на придвижване ще позволи измерване на неговите размери и в двете посоки.

Управлението на актуаторите обикновено става със същите контролери, които управляват моторизираните ролери, като може да бъде използван готов софтуер (plugin), ако актуатора е стандартен тип (с имплементирана вече логика от производителя на контролера) или да бъде зареден специфичен код написан от инженера интегратор на системата. Във вторият случай обикновено се предоставя графична среда за разработка на такъв софтуер т.н. ладер диаграма.



Фиг. 50 Завъртаща се маса



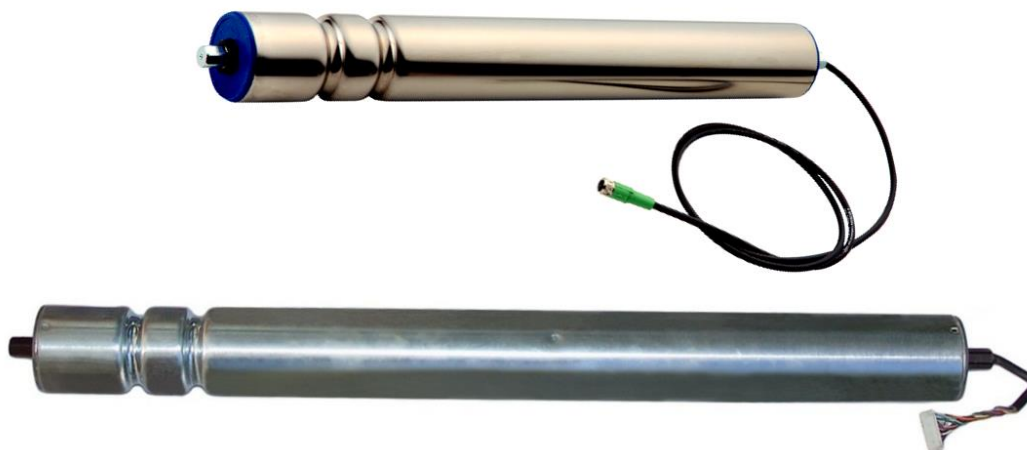
Фиг. 51 Устройство за напречен трансфер с повдигаци се ролки



Фиг. 52 Устройство за напречен трансфер с повдигаци се ремъци

III.5.2 Безчеткови постояннотокови мотори използвани за задвижване на конвейери

Безчетковите постояннотокови (БЧПТ) мотори се използват широко в конвейерните системи. С цел извличане на максимално количество полезна информация по време на тяхната работа те ще бъдат разгледани по-подробно.

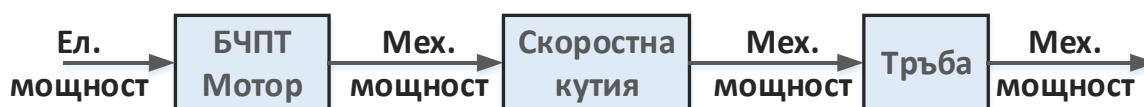


Фиг. 53 Типични моторизирани ролери

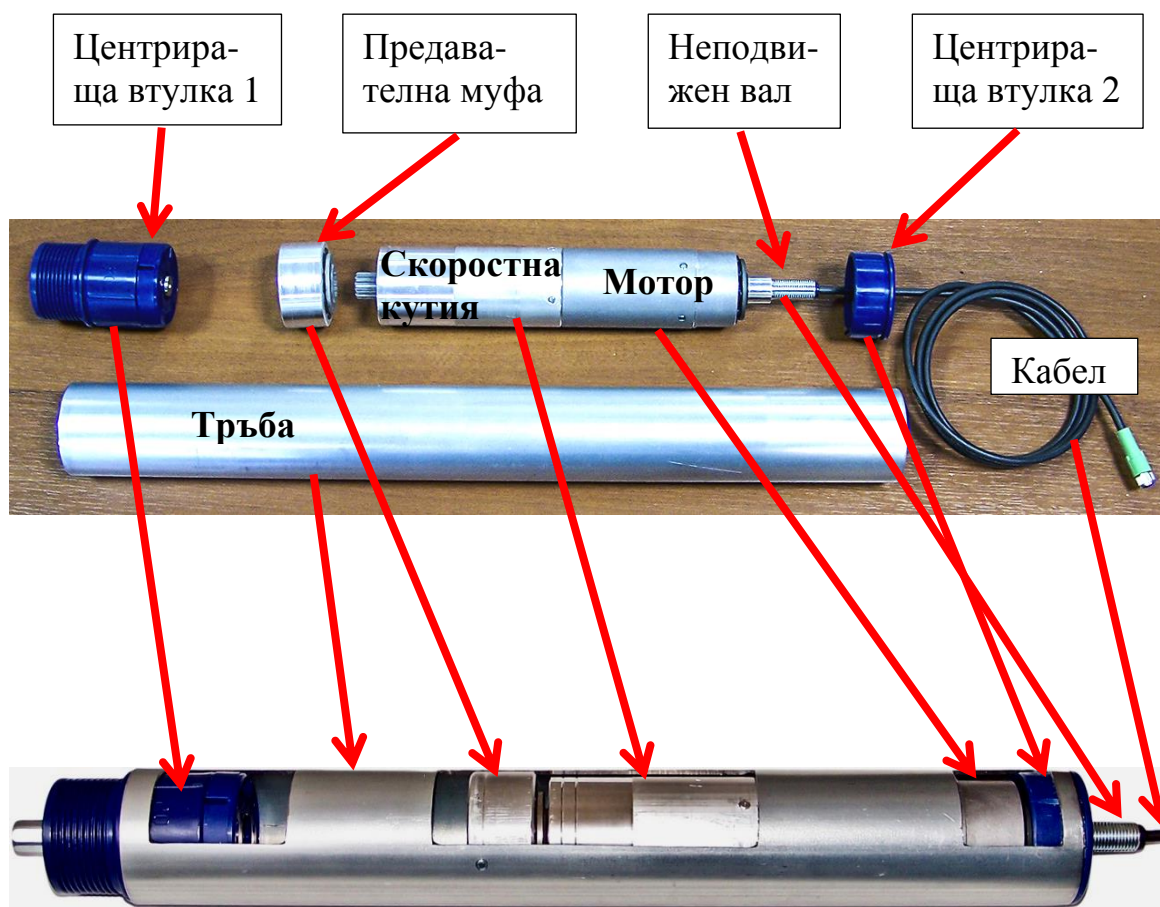
Фиг. 53 илюстрира изгледа на типични моторизирани ролери (MP) [18]. Те типично са със захранващо напрежение 24V или 48V, което е рамките на безопасното за конвейерни системи, където работят хора, но това не ограничава тяхното приложение само до такива системи. Те намират широко приложение и в среди недружелюбни с човешки персонал, като например с голяма чистота или пък много мръсни. Такива с много ниски температури (фризерни приложения) или високо наличие на някакво лъчение (рентгенови апарати за сканиране на багаж на летищата). Среди, които са непрекъснато мокри от някакви химикали и т.н.

Основните компоненти на MP са илюстрирани на Фиг. 54 до Фиг. 56:

- Мотор. Действа като задвижващ елемент. Той предава своя въртящ момент на тръбното тяло посредством скоростна кутия.
- Скоростна кутия. Тя редуцира ъгловата скорост на мотора съобразно предавателното си отношение.
- Центриращи втулки. Служат за центриране на модула „мотор-скоростна кутия“ към тръбното тяло.
- Предавателна муфа. Предава въртенето от модула „мотор-скоростна кутия“ към тръбното тяло.
- Тръбно тяло. Това е ротационния компонент на моторизирания ролер.



Фиг. 54 Блокова схема на MP



Фиг. 55 Основни компоненти на моторизиран ролер

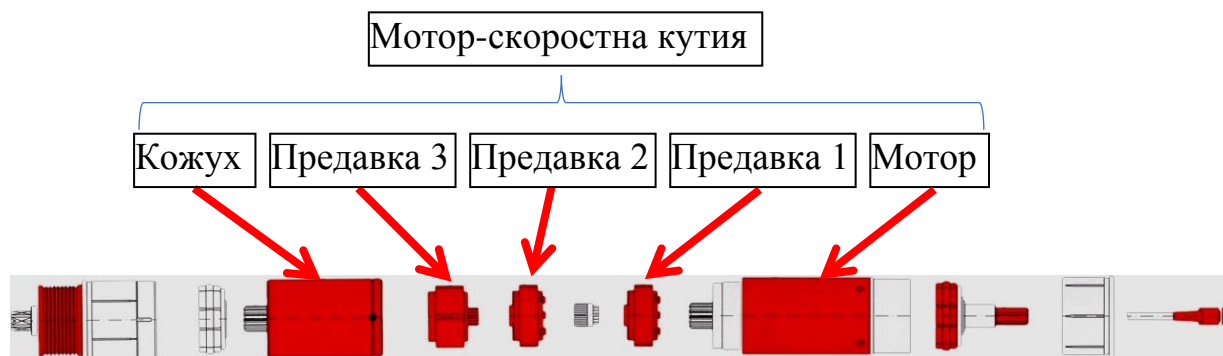
Моторът и скоростната кутия са обединени в един блок, обикновено наричан модул „мотор-скоростна кутия“ –Фиг. 56 и Фиг. 57, [18].

Вътрешният диаметър на тръбното тяло е по-голям от външния диаметър на блока „мотор-скоростна кутия“ за да се позволи независимо въртене на тръбното тяло. Изходният неподвижен вал на мотора, извеждащ изводите на статорната арматура е ротационно фиксиран към статора на мотора чрез специална абсорбираща муфа. Центриращата втулка 2 е пресована в тръбното тяло и осигурява центрирането на кухата ос, провеждаща входно-изходните кабели към арматурата на статора.

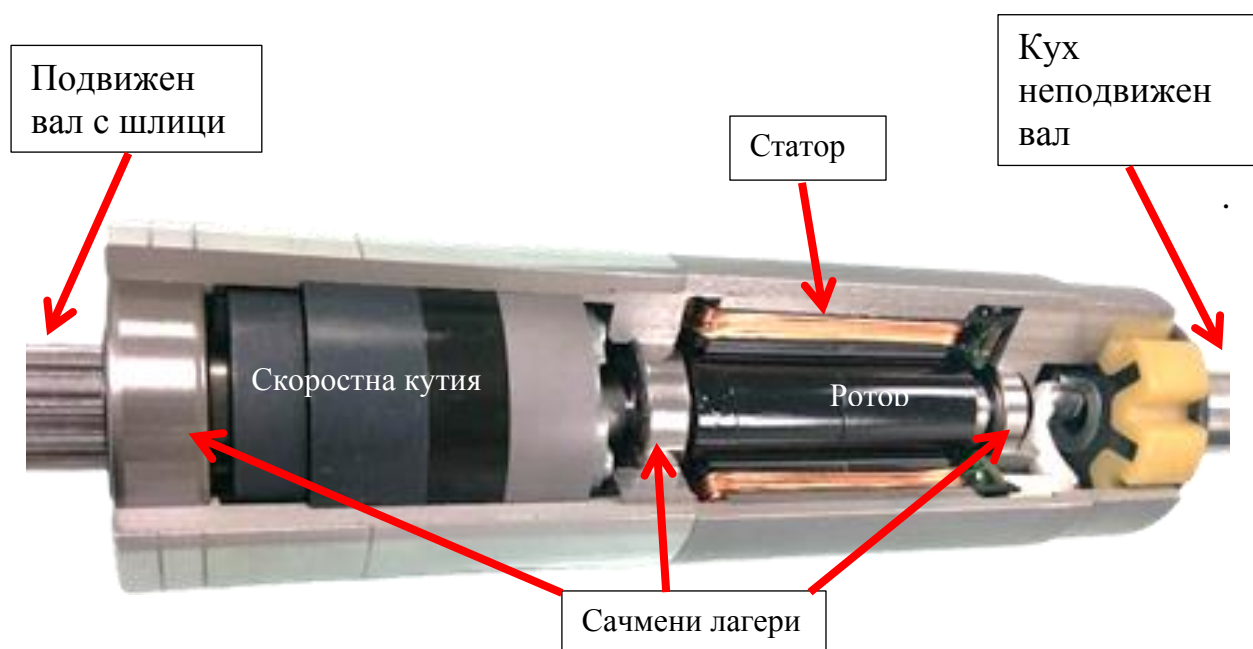
Предавателната муфа е занитена към тръбното тяло за да предава въртенето от модула „мотор-скоростна кутия“ към тръбното тяло. Центриращата втулка 1 е пресована в тръбното тяло и осигурява центрирането на въртящия се вал [18].

БЧПТ мотори притежават големи предимства, като добро съотношение момент към тегло, а също и висок КПД. Те са членове на голямото семейство на електрическите машини (Фиг. 58) и съчетават дълготрайността на индукционните мотори с линейността на ПТ мотори с

постоянни магнити. БЧПТ мотори предлагат широк обхват на скоростите, намалени размери и тегло, както и способност за развиване на голям въртящ момент.



Фиг. 56 Компоненти на мотор-скоростна кутия

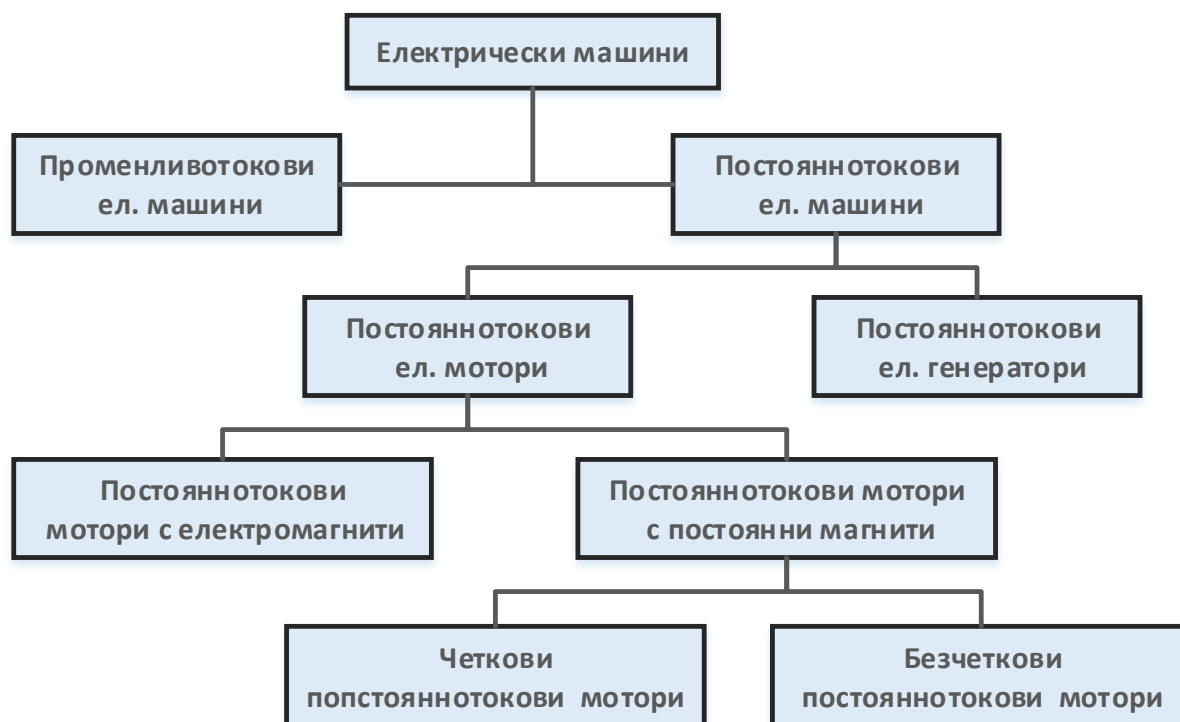


Фиг. 57 Разрез на БЧПТ двигател (С благодарност към Киьова Европа ЕООД)

На Фиг. 59 са показани двете главни съставни части на мотора - статора и ротора. Намотките са навити в статора, а постоянните магнити са разположени в ротора. Зъбното колело в края на задвижващия вал на ротора служи за зацепване на ротора към скоростната кутия [18].

Безчетковият постоянно-токов (БЧПТ) мотор прилича на четковия ПТ мотор по това, че и двата вида имат вътрешна обратна връзка по ъгловото положение на ротора. Тази обратна връзка подава информация кои намотки

от арматурата в кой точно момент да се включат. Поради тази обратна връзка двата типа мотори имат изключително важни свойства:

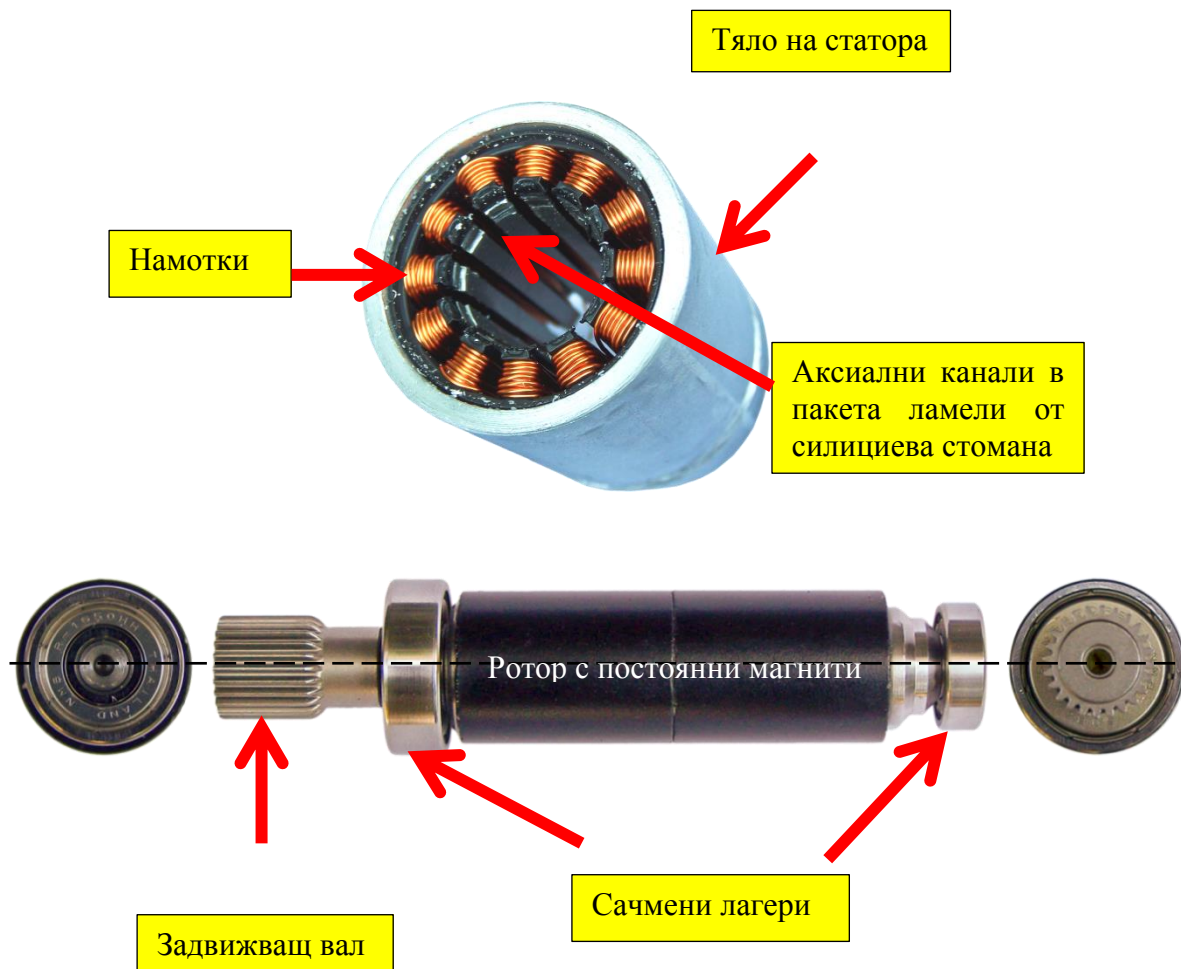


Фиг. 58 Опростена диаграма на семейството електрически машини

- Праволинейна зависимост „скорост / въртящ момент“.
- Висок стартов въртящ момент.
- Удобство за управление по скорост и по положение.

При четковите мотори обратната връзка се осъществява от механичен комутатор (изолирани медни пластини върху оста) и четки за подаване на захранващото напрежение.

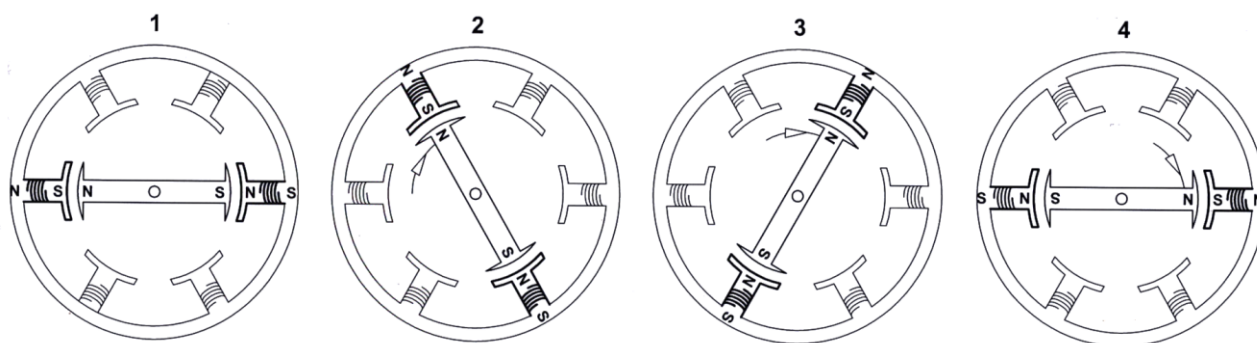
Обратната връзка при БЧПТ мотор се осъществява от безконтактни сензори за положение на оста, които подават информация към управляващата електроника. Тя (електронният блок) от своя страна комутира подходящата намотка в точно определен момент [16] , [17].



Фиг. 59 Статор и ротор на БЧПТ мотор

Характеристиките на БЧПТ мотори могат да се опишат накратко, както следва [45]:

- Въртящият момент е линейна функция на тока.
- Скоростта на въртене е линейна (обратно пропорционална) функция на въртящия момент, при константно захранващо напрежение.
- Скоростта на празен ход и въртящият момент при застопорен ротор са пропорционални на захранващото напрежение.
- Посоката на въртене се променя чрез начина на комутация от мостова схема на електронния блок.
- Способни са да работят в широк обхват на напрежение, скорост и въртящ момент.
- Работят при условия на ниски стойности на триене и на „магнитно зацепване“.
- Имат нисък термичен импеданс и висока мощност.
- Дълъг живот поради отсъствие на триещи се четки. Единственото триене е в лагерите (и като бъде куплиран - в скоростната кутия).
- Имат много по-малко електромагнитно излъчване, поради отсъствието на комутиращи четки.



Фиг. 60 Илюстрация на принципа на работа на БЧПТ мотор

Фиг. 60 илюстрира принципа на работа на БЧПТ мотор, основан на физическите явления на привличане на разноименните и отблъскване на едноименните магнитни полюси [18].

Постоянните магнити са монтирани върху ротора. Три двойки намотки, формиращи три електромагнита, са навити в специални канали на статора. Следвайки сигналите на сензорите за положението на оста, всяка двойка бобини се активира в такава последователност и момент че да се осигури устойчиво въртливо движение на ротора. Процесът на превключване на бобините се нарича **комутация**.

Ако разгледаме подробно Фиг. 60:

- **Позиция 1.** Включени са хоризонталните бобини, които формират магнитни полюси с поляритет, показани на фигурата. Другите две двойки бобини НЕ са включени. Постоянният магнит се насочва хоризонтално, поради силата на привличането.

- **Позиция 2.** Хоризонталните бобини се изключват, а се включват съседните бобини, които формират магнитни полюси с показаня поляритет. Другите две двойки бобини НЕ са включени. Постоянният магнит се завърта на дясно по часовата стрелка, поради новата сила на привличане.

- **Позиция 3.** Горните бобини се изключват, а се включват съседните бобини, които формират магнитни полюси с показаня поляритет. Другите две двойки бобини НЕ са включени. Постоянният магнит се завърта на дясно по часовата стрелка, поради новата сила на привличане.

- **Позиция 4.** И така, отново и отново, предишните бобини се изключват, а се включват съседните бобини, които формират нови магнитни полюси. Другите две двойки бобини НЕ са включени. Постоянният магнит се завърта на дясно по часовата стрелка, поради новата сила на привличане и т.н. до достигане на позиция 6, от която следваща е отново позиция 1.

Процесът на последователно превключване на бобините се осъществява от електронен модул за управление (контролер на зоната на конвейера), който може да подаде различен ток през намотките в зависимост от необходимостта от ускоряване или спиране.

Броят на магнитите, броят на бобините, както и тяхната структура варират с оглед повишение на коефициента на полезно действие на мотора. Винаги броят на северните и южните полюси е еднакъв и броя на тези двойки се означава с 'р'. Когато $p > 1$ се получава случай в който едно електрическо завъртане не е равно на едно механично завъртане на вала (ротора). Или казано иначе контролера трябва да направи р електрически завъртания (р броя последователности 1,2,3,4,5,6), за да се осъществи завъртане на ротора с един механичен оборот. Ако означим броя преходи между състоянията на комутация с n следва (14)

$$n_{mr} = pn_{er} \quad (14)$$

Където:

n_{mr} е броя преходи за едно механично завъртане на ротора.

n_{er} е броя преходи за едно електрическо завъртане на ротора, който за описания метод на комутация е 6.

p е броя двойки полюси на ротора

Скоростната кутия има предавателно число даващо връзката между броя завъртания на вала на мотора и броя завъртания, които ще направи тръбата (тялото на ролера) (15). Това число е произведението от предавателните отношения на всички предавки от скоростната кутия. То се избира според желаната скорост на въртене на ролера и тази при която моторът има максимален КПД.

$$i = \frac{\omega_n}{\omega_{req}} \quad (15)$$

Където:

i е предавателното число на редуктора

ω_{req} е желаната работна скорост на ролера

ω_n е скоростта на мотора, при която той работи с максимален КПД.

Ако интегрираме (15) по времето получаваме (16)

$$i = \frac{r_{mr}}{r_{mo}} \quad (16)$$

Където:

r_{mr} е броя механични завъртания на мотора (ротора му)

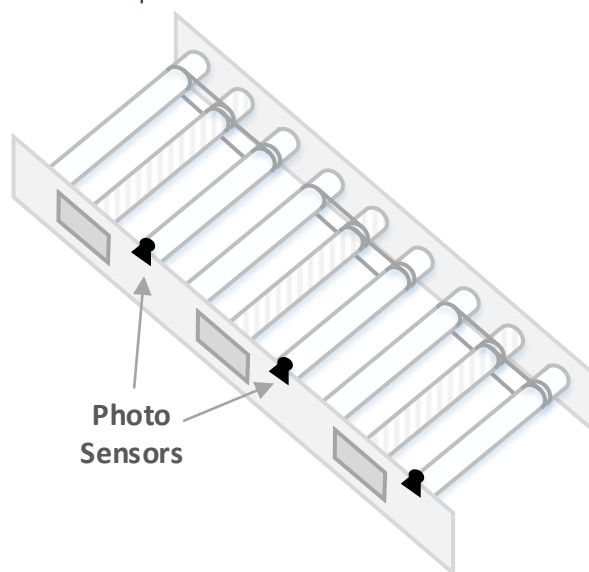
r_{mo} е броя механични завъртания на ролера

Използвайки (15) и (16) и имайки предвид, че $n_{er} = 6$, може да се изведе (17):

$$n_{mo} = 6pi \quad (17)$$

Типичен моторизиран ролер има 5 двойки полюси и предавателно число на скоростната кутия 15. Според (17) това води до 450 броя комутации за един оборот на ролера. Това позволява точност на позициониране от 0.8° . Алгоритъма за управление на зоната от по-високо ниво може да използва тази информация заедно с информацията от фотосензора за по-точно позициониране на обекта върху зоната.

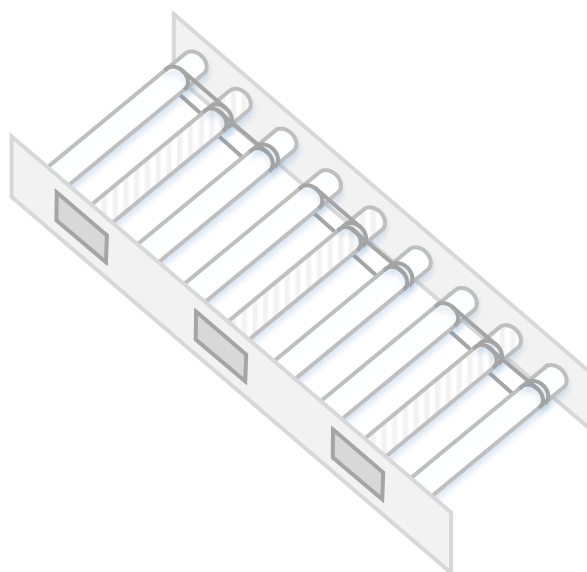
III.5.3 Безсензорни конвейери



Фиг. 61 Конвейер с фотосензори

По време на въртене на БЧПТ мотор се извършва непрекъснато измерване на стойността на тока през включените двойки намотки с цел управление на средното подадено към мотора напрежение и предпазването му от претоварване. При начално стартиране обикновено се допуска големината на този ток да е над номиналните стойности за мотора, но това е само за кратко [46].

Информацията за големината на този ток може да бъде използвана за оценка на наличието на обект върху зоната в случаите, когато теглото на обектите не е по-малка от някаква стойност, която се определя от типа на ролерите (предавателното число на скоростните им кутии) и минималният ток, който може да бъде надеждно измерен от контролера. В тези случаи зоната може да функционира и без фотосензор за отчитане на наличието на обект, което води до значително намаляване на себестойността на зоната и конвейера като цяло Фиг. 61 и Фиг. 62. Снимка на истински, конвейер, който беше направен експериментално и осъществяващ безсензорно установяване на позицията на мотора и безсензорно определяне на наличието на обект е показана на Фиг. 63 [47], [48].



Фиг. 62 Безсензорен конвейер

Втора възможност за премахването на фотосензора е използването на броя комутации на мотора направени след сигнала за подаване на обект от предишната зона. Това изисква предварително конфигуриране на дължината на зоната като брой комутации за всички зони, които няма да имат фотосензори. Обикновено това не представлява голяма трудност, защото се използва конфигурационен софтуер и поради това, че обикновено повечето зони са с еднаква дължина.

Зоната придвижва обекта за определения ѝ при конфигурирането брой комутации и подава сигнал на следващата зона за наличие на обект, който тя трябва да поеме. Контролно, през определен брой безсензорни зони се поставя една оборудвана с фотосензори, която да извърши евентуално синхронизиране на позицията на обектите, ако те са натрупали грешка от приплъзване. При правилно конфигурирана система (ускорения и забавяния съобразени с материала и сцеплението на обектите с моторизираните ролери) може да има значително голям брой последователни безсензорни зони, Фиг. 63 [49], [48].



Фиг. 63 Безсензорен конвейер с безсензорни БЧПТ мотори.

III.5.4 Приложение на предложения метод

При използването на този алгоритъм, конвейерът ще оставя големи празни разстояние между товарите и ще се опитва да препраща бързо, когато товарите са малко на брой. Ако скоростта, с която постъпва товарите се увеличи, то разстоянията между товарите ще станат по-малки, което води до по-добра плътност на съхранение, както е показано на Фиг. 64 и Фиг. 65.



Фиг. 64 Конвейер с малка плътност на обектите



Фиг. 65 Конвейер с голяма плътност на обектите

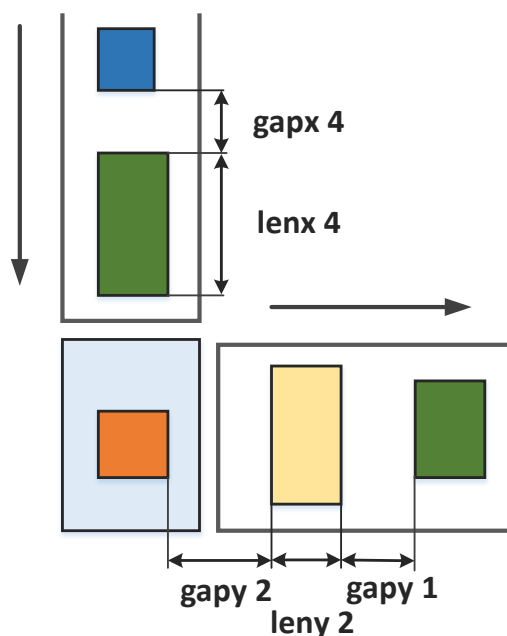
Предложеният алгоритъм е подходящ за конвейерни приложения, при които движението на товара назад е нежелано или непозволено. Ако обаче конвейерът позволява такова движение, то зоните могат допълнително да намалят празните разстояния като придвижат товарите назад, докато се достигне зададеният размер. Това допълнение към алгоритъма трябва да се

осъществява с хистерезис между необходимата и действителната разлика в разстоянията, тъй като при навлизането на нови товари в конвейера зоните ще преизчислят разстоянията като по-ниски. Това ще доведе до движения целящи да намалят празните разстояния при всеки нововъведен товар, което не е енергийно ефективно и не се оправдава.

Зоните могат автоматично да изчисляват стойността на хистерезиса, така че движението на всички зони, които могат да намалят празните разстояния, ще се случи, когато местата (свободни зони) в горната част на конвейера са по-малко или равно на две. Това ще гарантира, че линейната част няма да блокира своя източник и ще съхранява възможно най-много товари.

Обикновено източникът на товарите е система или конвейер на трета страна, чиито комуникационния протокол е различен от този реализиран в линейната част. Той предава товарите, а линейната част ги приема, използвайки някои от известните индустриални протоколи (Modbus TCP, Profinet, Ethercat CC-Link, EthernetIP и др.)

Това прави невъзможно първата зона да знае каква е дължината на входящия товар и големината на празното разстояние до следващия товар. Дори ако цялата система поддържа гореспоменатия протокол и алгоритъм, има участъци където се осъществява Right Angled Transfer (RAT), показани на Фиг. 66, които извършват перпендикулярни движения на товара.



Фиг. 66 Конвейер с перпендикулярно придвижване

Тези устройства често са разположени в ъглите, вместо в крива секция. Това обаче не може да се счита за край на линейната част, тъй като не

възпрепятства товарния поток. Измерените празни разстояния се обезсмислят след такова придвижване, защото са подготвени за посока, която е перпендикулярна на движението.

IV ГЛАВА. ЕКСПЕРИМЕНТИ И АНАЛИЗ НА РЕЗУЛТАТИТЕ

IV.1 Експерименти с предложеният метод за подобряване качеството на данните за позицията на моторите

Следният лог файл показва направените корекции на позициите на ротора. Например на първи ред, грешки, налични и в трите блока три битови полета са коригирани. На база на списъка с очаквания „ExpHall:6,2,3“, който е подреден в низходящ ред по вероятност на очакването.

```
MotMain:[81819028]Old(hall:2, pos:5), ExpHall:6,2,3, New(hall:6, pos:0), RX:[00|4|7][10|3|6]<-[10|2|2][10|2|2]<-[10|3|3][10|3|3]
MotMain:[81819760]Old(hall:2, pos:5), ExpHall:2,6,3, New(hall:2, pos:5), RX:[10|2|2][00|4|4]<-[10|2|2][10|2|2]<-[10|2|2][10|2|2]
MotMain:[81820636]Old(hall:2, pos:5), ExpHall:2,6,3, New(hall:2, pos:5), RX:[10|2|2][00|4|4]<-[10|2|2][10|2|2]<-[10|2|2][10|2|2]
MotMain:[81820796]Old(hall:2, pos:5), ExpHall:2,6,3, New(hall:2, pos:5), RX:[00|4|7][10|3|6]<-[10|2|2][10|2|2]<-[10|2|2][10|2|2]
MotMain:[81820836]Old(hall:2, pos:5), ExpHall:2,6,3, New(hall:2, pos:5), RX:[10|2|2][11|7|6]<-[10|2|2][10|2|2]<-[00|4|7][10|3|6]
MotMain:[81821000]Old(hall:2, pos:5), ExpHall:2,6,3, New(hall:2, pos:5), RX:[10|2|3][10|3|6]<-[10|2|2][10|2|2]<-[10|2|2][10|2|2]
MotMain:[81821812]Old(hall:2, pos:5), ExpHall:2,6,3, New(hall:2, pos:5), RX:[10|2|2][00|4|4]<-[10|2|2][10|2|2]<-[10|2|2][10|2|2]
MotMain:[81825060]Old(hall:2, pos:5), ExpHall:2,6,3, New(hall:2, pos:5), RX:[00|4|7][10|3|6]<-[10|2|2][10|2|2]<-[10|2|2][10|2|2]
MotMain:[81827948]Old(hall:2, pos:5), ExpHall:2,6,3, New(hall:2, pos:5), RX:[10|2|2][11|7|6]<-[10|2|2][10|2|2]<-[10|2|2][10|2|2]
MotMain:[81828028]Old(hall:2, pos:5), ExpHall:2,6,3, New(hall:2, pos:5), RX:[10|3|6][11|6|2]<-[10|2|2][10|2|2]<-[10|2|2][10|2|2]
MotMain:[81828984]Old(hall:2, pos:5), ExpHall:2,6,3, New(hall:2, pos:5), RX:[10|3|6][11|6|2]<-[10|2|2][10|2|2]<-[10|2|2][10|2|2]
MotMain:[81829352]Old(hall:2, pos:5), ExpHall:2,6,3, New(hall:2, pos:5), RX:[10|2|2][11|7|6]<-[10|2|2][10|2|2]<-[10|2|2][10|2|2]
MotMain:[81830632]Old(hall:2, pos:5), ExpHall:2,6,3, New(hall:2, pos:5), RX:[10|3|7][11|6|2]<-[10|2|2][10|2|2]<-[10|2|2][10|2|2]
MotMain:[81832664]Old(hall:2, pos:5), ExpHall:2,6,3, New(hall:2, pos:5), RX:[00|6|6][10|2|2]<-[10|2|2][10|2|2]<-[10|2|2][10|2|2]
MotMain:[81833676]Old(hall:2, pos:5), ExpHall:2,6,3, New(hall:2, pos:5), RX:[10|3|7][11|6|2]<-[10|2|2][10|2|2]<-[10|2|2][10|2|2]
MotMain:[81833860]Old(hall:2, pos:5), ExpHall:2,6,3, New(hall:2, pos:5), RX:[10|3|7][11|6|2]<-[10|2|2][10|2|2]<-[10|2|2][10|2|2]
MotMain:[81834248]Old(hall:2, pos:5), ExpHall:2,6,3, New(hall:2, pos:5), RX:[10|3|6][11|6|2]<-[10|2|2][10|2|2]<-[10|2|2][10|2|2]
MotMain:[81837072]Old(hall:2, pos:5), ExpHall:2,6,3, New(hall:2, pos:5), RX:[00|4|4][10|2|2]<-[10|2|2][10|2|2]<-[10|2|2][10|2|2]
```

```

MotMain:[81837336]Old(hall:2, pos:5), ExpHall:2,6,3, New(hall:2,
pos:5), RX:[10|2|2][00|4|6]<-[10|2|2][10|2|2]<-[10|2|2][10|2|2]
MotMain:[81838108]Old(hall:2, pos:5), ExpHall:2,6,3, New(hall:2,
pos:5), RX:[00|7|7][11|6|2]<-[10|2|2][10|2|2]<-[10|2|2][10|2|2]
MotMain:[81839508]Old(hall:2, pos:5), ExpHall:2,6,3, New(hall:2,
pos:5), RX:[10|2|3][11|7|6]<-[10|2|2][10|2|2]<-[10|2|2][10|2|2]
MotMain:[81839548]Old(hall:2, pos:5), ExpHall:2,6,3, New(hall:2,
pos:5), RX:[10|2|2][11|7|6]<-[10|2|2][10|2|2]<-[10|2|3][11|7|6]
MotMain:[81840052]Old(hall:2, pos:5), ExpHall:2,6,3, New(hall:2,
pos:5), RX:[00|0|2][00|0|2]<-[00|0|0][00|0|0]<-[00|0|3][00|0|0]
MotMain:[81840128]Old(hall:2, pos:5), ExpHall:2,6,3, New(hall:8,
pos:255), RX:[00|0|1][00|1|5]<-[00|0|2][00|0|2]<-[00|0|0][00|0|0]
MotMain:[81840232]Old(hall:2, pos:5), ExpHall:2,6,3, New(hall:2,
pos:5), RX:[00|4|7][10|3|6]<-[10|2|2][10|2|2]<-[10|2|2][10|2|2]
MotMain:[81840476]Old(hall:2, pos:5), ExpHall:2,6,3, New(hall:2,
pos:5), RX:[10|3|7][11|6|2]<-[10|2|2][10|2|2]<-[10|2|2][10|2|2]
MotMain:[81842576]Old(hall:2, pos:5), ExpHall:2,6,3, New(hall:2,
pos:5), RX:[00|0|2][00|0|2]<-[00|0|0][00|0|0]<-[00|0|1][00|0|0]
MotMain:[81842656]Old(hall:2, pos:5), ExpHall:2,6,3, New(hall:8,
pos:255), RX:[00|0|1][00|1|5]<-[00|0|2][00|0|2]<-[00|0|0][00|0|0]
MotMain:[81842796]Old(hall:2, pos:5), ExpHall:2,6,3, New(hall:2,
pos:5), RX:[10|2|3][10|3|6]<-[10|2|2][10|2|2]<-[10|2|2][10|2|2]
MotMain:[81842876]Old(hall:2, pos:5), ExpHall:2,6,3, New(hall:2,
pos:5), RX:[10|2|2][11|7|6]<-[10|2|2][10|2|2]<-[10|2|2][10|2|2]

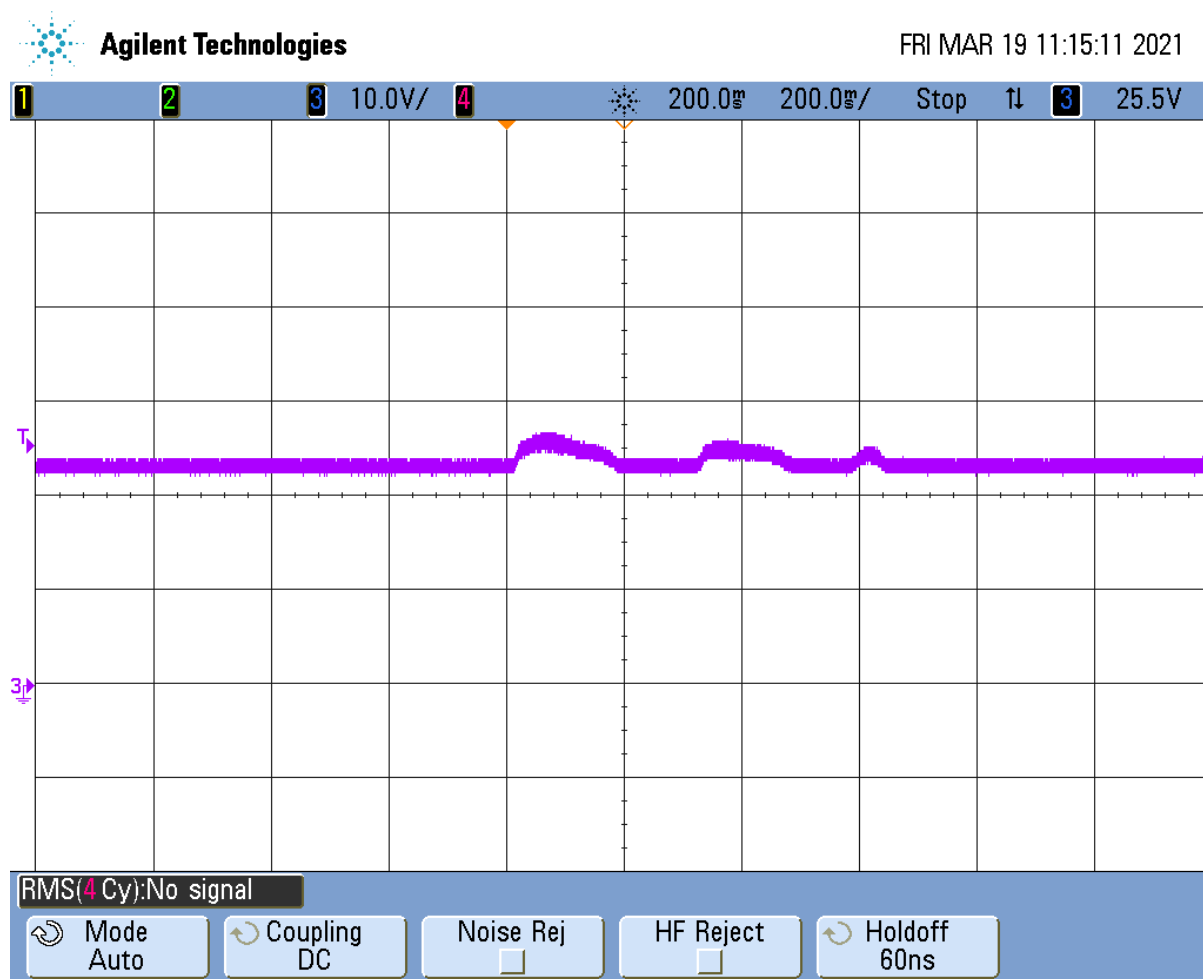
```

Благодарение на предложения метод успешно се коригират грешки в получената информация (предава се по линия в общ кабел със силовите проводници) за положението на ротора, с което се подобрява на работа на моторизирания ролер.

IV.2 Експерименти с предложения метод за подобряване на захранването

На Фиг. 42 и Фиг. 43 са показани случаи на връщане на енергия към захранващия източник при работата на типичен контролер за управление на зона на конвейер, при спиране на различни по тежест обекти. В случая на по-тежък обект, които е натрупал значително по-голяма кинетична енергия при спирането си повишението е също значително. Номиналното захранващо напрежение е 24V, а вертикалната резолюция на осцилограмата е 10V/деление. От графиките се вижда, че при по-тежък обект нарастването може да достигне до 40V, което е почти двойно на номиналната стойност. Това е прага на дефектиране на електронните компоненти в контролера и захранващия блок.

На Фиг. 67 е показана графика от спирането на същите обекти, но при интелигентно разпределяне на енергията, при който забавянето на спиращи обекти се ограничава, докато останалите зони, консумиращи енергия 'погълнат' върнатото количество.



Фиг. 67 Осцилограма на напрежението при спиране с използване а новия метод за разпределение на енергията

Проучването, разработването и имплементацията на метода за установяване и разпределение на консумираната енергия от захранващите източници са представени на международната конференция IEEE-IS'2020:

Andonov, I, Tsvetanov, S, Power supply map discovering and efficient energy distribution method for distributed conveyor systems, IEEE-IS'2020 Conference, Varna, Bulgaria, pages: 1541-1672, Sep. 2020, DOI: 10.1109/IS48319.2020.9200130.

IV.3 Експерименти с предложения метод за разпределено управление

IV.3.1 Очаквани резултати

Конвейерът, използващ този алгоритъм има големи изчислени празнини и добра пропускателна способност, ако броят на товарите между началото и края е малък. Ако броят на прииждащите товари се увеличи, то празнините между скоро постъпилите товари ще се стеснят, което ще позволи по-добра плътност на съхранение Фиг. 69



Фиг. 68 Ниска плътност на обектите

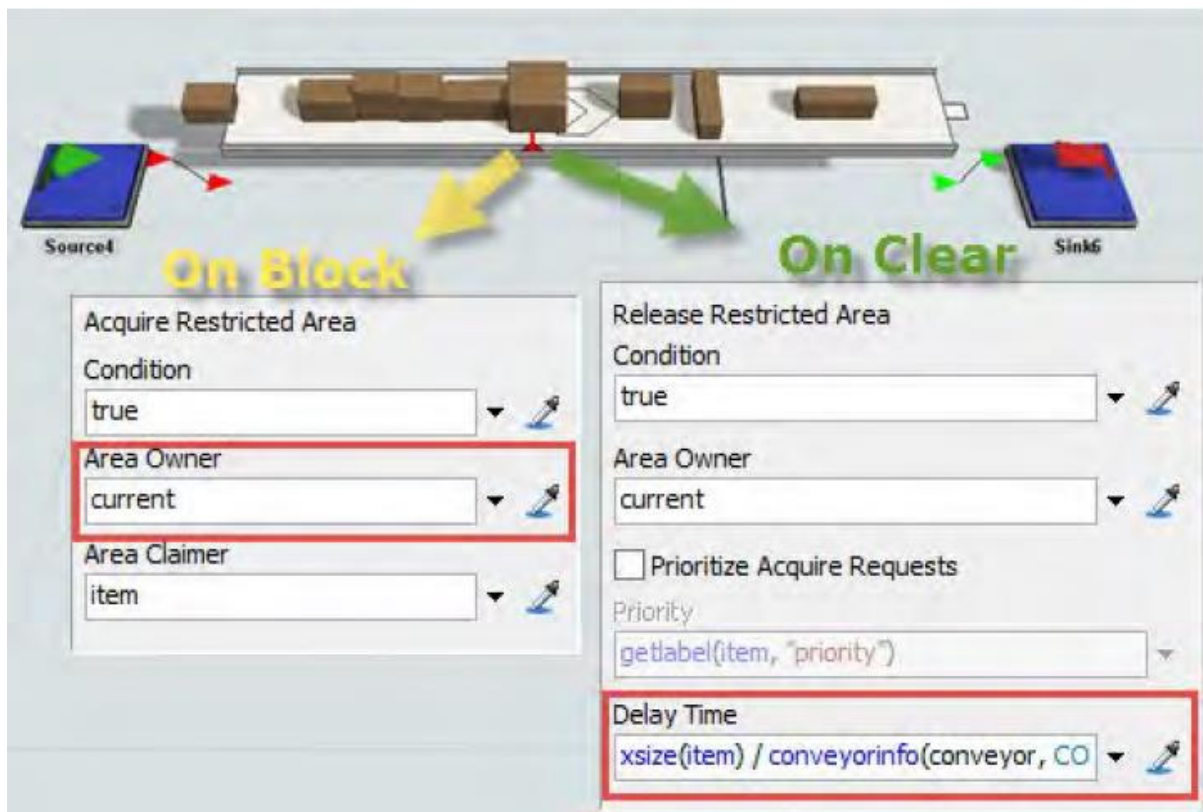


Фиг. 69 Висока плътност на обектите

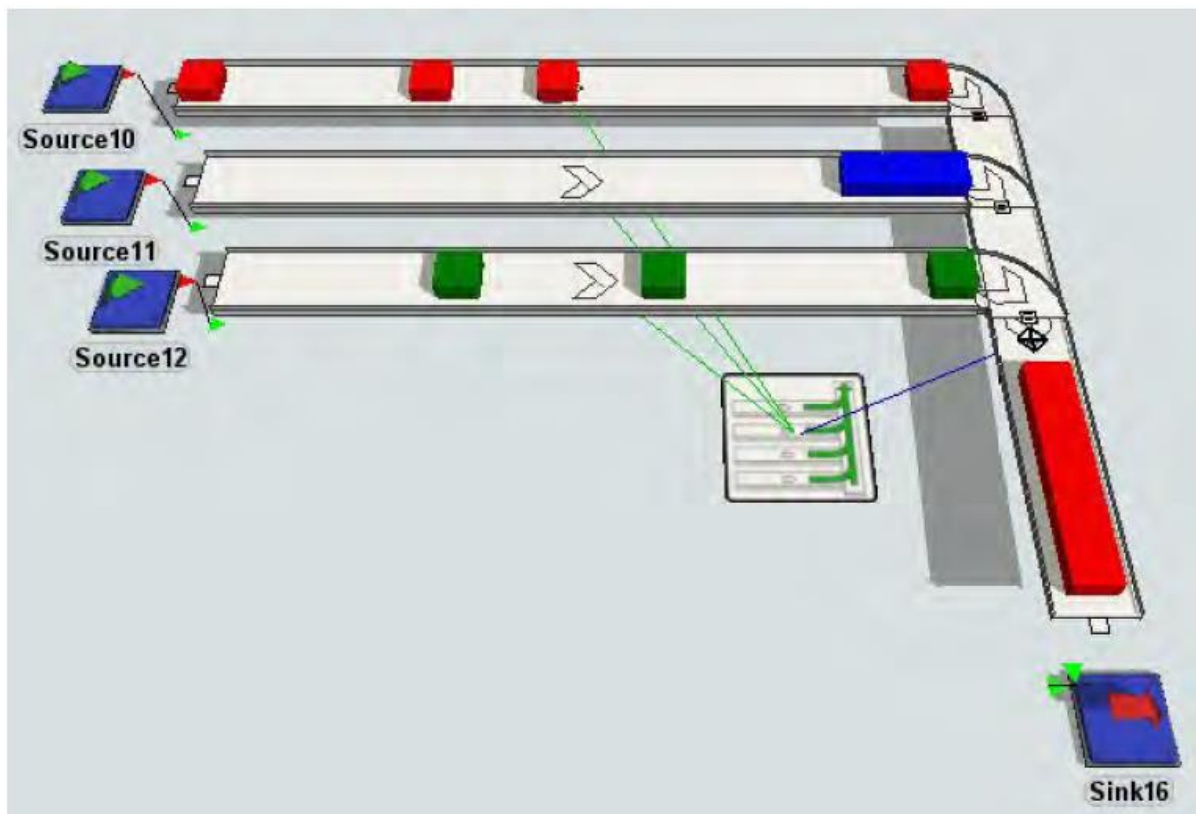
Зоните могат да продължат измерването на размерите на натоварите и празнините и да правят статистическо усредняване на стойността, с цел да я доближат до реалната. Заедно със стойността за всеки размер (на товара или празнината), зоните могат да обменят допълнителна информация, показваща колко вероятно е тази стойност да представлява реалния размер. Ако дадена зона установи, че по време на нейното статистическо усредняване стойността е много по-различна от получената, която има по-висок коефициент на надеждност (което означава, че много зони са доказали тази стойност като правилна), може да приеме, че има някаква хлъзгава или външна намеса над товара, Фиг. 8.

IV.3.2 Резултати от симулациите

Извършени са компютърни симулации, чрез които се доказва работоспособността на предложения алгоритъм и метод. Използван е софтуер за компютърни симулации Flexsim.



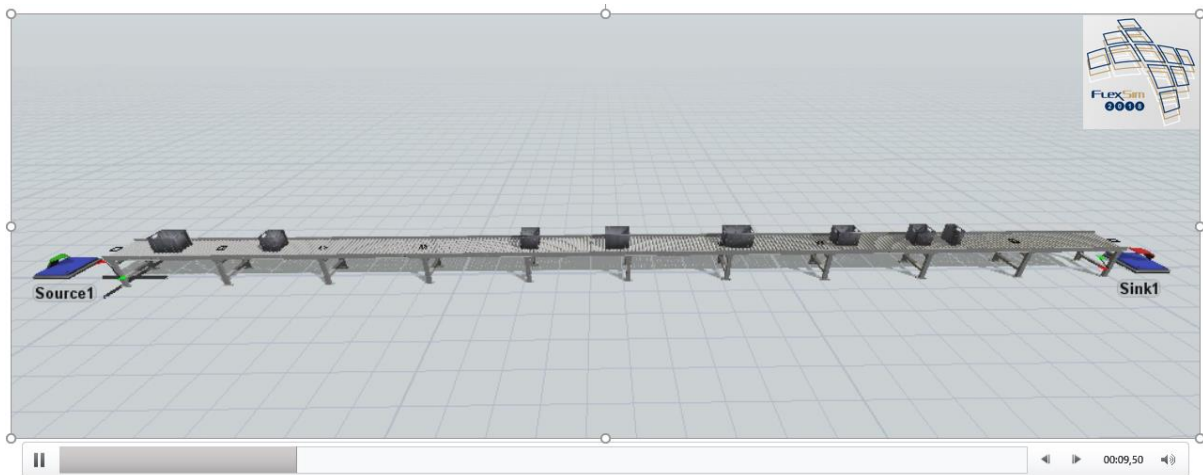
Фиг. 70 Конфигуриране на симулацията



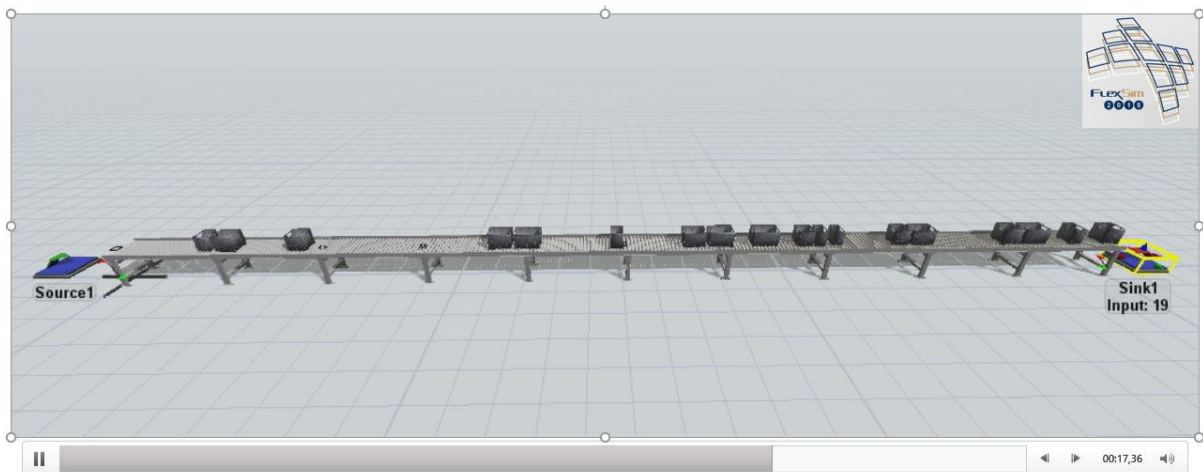
Фиг. 71 Симулиране на конвейер със различни по размер обекти

Резултатите от симулациите показват повишаване на ефективността на системата, което може да се постигне чрез сравнително прости разпределени изчисления, което позволява използването на съществуващия хардуер.

Повишаването на ефективността на системата се постига без необходимост от допълнителни настройки



Фиг. 72 Работа на конвейера без акумулиране на обектите



Фиг. 73 Акумулация на конвейера след преминаването на 20 обекта

Проучването, разработването и имплементацията на метода за разпределено управление на конвейерни системи базиран на софтуерни агенти е представен на международната конференция IEEE-IS'2018:

Andonov, I, Tsvetanov, S, **A Novel Algorithm for Distributed Control of Conveyor Systems**, IEEE-IS'2018 Conference, Madeira Island, Portugal, pages: 379-372, Sep. 2018, DOI: 10.1109/IS.2018.8710518

IV.4 Експерименти и тестове на цялата система

Към предложеното решение има интерес от бизнеса и ни бе предоставена възможност да проведем експерименти върху реална, макар и сравнително проста, конвейерна система Фиг. 74.



Фиг. 74 Експериментален конвейер

Компютърните симулации, както и проведените експерименти дават обнадеждаващи резултати.

Същинското предимство на разработения метод за децентрализираното управление се проявява при имплементиране при много по-сложен конвейер Фиг. 75 и Фиг. 76.



Фиг. 75 Входяща част от конвейер в център за сортиране на пратки



Фиг. 76 Рециркулираща част от конвейер

IV.5 Заключение

Благодарение на предложения иновативен метод за разпределено управление на конвейерни системи, който подобрява надеждността и производителността.

Разпределеният алгоритъм буферира възможно най-много товари върху конвейера, като същевременно запазва пропускателната способност висока, като за тази цел се осъществява комуникация между зоните и се предава информация за разстоянието, на което трябва да се преместят товарите.

По този начин зоната може динамично да удължи или скъси празнините между товарите, което благоприятства за по-добра пропускателна способност и по-добра плътност.

Извършено е изследване и анализ на методите и средствата за управление на конвейерни системи, като е предложено решение за подобряване ефективността на конвейерни системи, чрез прилагане на нови методи за разпределено управление

Извършено е моделиране и компютърно симулиране на предложения метод, а също така е проектиран и имплементиран прототип на конвейерна система, чрез които да се валидира работоспособността на метода

Разработени са тестови сценарии за верификация и оценка на ефективността на метода, които потвърждават постигнатото ниво на подобряване на надеждността и производителността

Предложеният метод за подsigуряване на контролерите срещу клониране е представен на 9th Balkan Conference on Informatics, 2019.

Проучването, разработването и имплементацията на метода за установяване и разпределение на консумираната енергия от захранващите източници са представени на международната конференция IEEE-IS'2020.

Проучването, разработването и имплементацията на метода за разпределено управление на конвейерни системи базиран на софтуерни агенти е представен на международната конференция IEEE-IS'2018.

Като бъдещо развитие се предвижда създаването на метод за автоматично конфигуриране на параметрите, чрез използване на машинно обучение. Ако системата открие регулярна промяна на натоварването в определен част от денонощието, ден от седмицата и т.н., ще може да се самообучи и да промени параметрите си, с цел да придвижва товарите по-ефективно.

Приноси

- Извършено е изследване и анализ на методите и средствата за управление на конвейерни системи;
- Предложено е решение за подобряване ефективността на конвейерни системи, чрез прилагане на нов метод за разпределено управление;
- Предложени и имплементирани са нови методи подобряващи работата на подсистемите за измерване, захранване и комуникация;
- Извършено е моделиране и компютърно симулиране на предложения метод за разпределено управление;
- Проектиран е и е имплементиран прототип на конвейерна система, чрез който да се валидира работоспособността на метода;
- Разработени са тестови сценарии за верификация и оценка на ефективността на метода, които потвърждават постигнатото ниво на подобряване на надеждността и производителността.

Публикации

- Andonov, I, Tsvetanov, S, **A Novel Algorithm for Distributed Control of Conveyor Systems**, IEEE-IS'2018 Conference, Madeira Island, Portugal, pages: 379-372, Sep. 2018, DOI: 10.1109/IS.2018.8710518
- *Ivailo Andonov, Simeon Tsvetanov, Stefan Dimitrov*, **Securing IoT devices against cloning**, COMPUTER & COMMUNICATIONS ENGINEERING, vol:13, issue:2, 2019, pages:25-28, ISSN (print):1314-2291
- *Андонов, И.*, **Разпределено управление на конвейерни системи**, ISI Journal, issue: 1, 2020, ISSN 2534-8531, (под печат)
- *Andonov, I, Tsvetanov, S*, **Power supply map discovering and efficient energy distribution method for distributed conveyor systems**, IEEE-IS'2020 Conference, Varna, Bulgaria, pages: 1541-1672, Sep. 2020, DOI: 10.1109/IS48319.2020.9200130

ИЗПОЛЗВАНА ЛИТЕРАТУРА

- [1] R. Rosen, *Anticipatory Systems*, Oxford, MA: Pergamon Press, 1985.
- [2] D. K. W. K. Robert Król, „Determination of The Mechanical Power in Belt Conveyor’s Drive System in Industrial Conditions,“ в *World Multidisciplinary Earth Sciences Symposium*, 2016.
- [3] D. A. R. S. Pradnyaratna A Meshram, „Design, Modeling and Analysis of conveyor system used for transportation of Cartons,“ *International Journal of Research in Advent Technology*, том 4, № 1, January 2016.
- [4] C.-H. Lan, „The design of a multi-conveyor system for profit maximisation,“ *The International Journal of Advanced Manufacturing Technology*, том 22, p. 510–521, 2003.
- [5] V. M. P. N. k. A. V. R. L. K. P. S., „Type of Conveyor System: A Review,“ *International Journal for Scientific Research & Development*, том 2, № 12, 2015.
- [6] D. V. Hall, „Control system for an accumulating conveyor“. Worldwide Патент US5228558A, 23 12 1991.
- [7] G. S. S. Pratik P. Gandhi, „The Pallet Conveyor System Application In The Industrial Lines - A New Design System With Improvement Of Productivity,“ *International Journal of Mechanical and Production Engineering Research and Development (IJMPERD)*, том 9, № 3, pp. 257-266, Jun 2019.
- [8] H. M. K. W. Gładysiewicz L., „Determining belt resistance,“ *Bulk Handling Today*, том 5, № 1, pp. 23-28, 2009.
- [9] K. W. K. D. G. L. Król R., „Testing belt conveyor resistance to motion in underground mine conditions,“ *International Journal of Mining, Reclamation and Environment*, pp. 1-13, 2016.
- [10] L. Z. X. X. Mukalu S. Masaki, „Cost optimization design approach for multiple drive belt conveyors,“ *36th Chinese Control Conference*, 2017.
- [11] S. Ahmed, *Position Sensorless and Energy-efficient Electromagnetic Levitation for Translational Motion Conveyor System. Smart use of high frequency switching noise*, Tokyo: University of Tokyo, 2016.

- [12] J. Antoniak, „Theoretical basis and industrial applications of energy – saving and increased durability belt conveyors,“ *Acta Montan*, том 8, p. 106–117, 2003.
- [13] J. B. Vojtěch Vigner, „Precise synchronization in large distributed systems,“ в *IEEE 7th International Conference on Intelligent Data Acquisition and Advanced Computing Systems (IDAACS)*, Berlin, Germany, 2013.
- [14] R. d. J.-M. J.-R. G.-E. J. R. G. d. M. J. M. B.-A. Francesc D Muñoz-Escóí, „CAP Theorem: Revision of Its Related Consistency Models,“ *The Computer Journal*, том 62, № 6, pp. 943 - 960, 2019.
- [15] B. D. Clénet, *Optimising energy efficiency*, Rueil Malmaison - France: Schneider Electric SA, January 2010.
- [16] A. D. P. Markos, *An Index that Correlates Service Performance and Energy Consumption of Belt Conveyors*, *FME Transactions* 46, 313-31, 2018.
- [17] C. M. . L. Jianhua Ji, „Research on the energy-saving control strategy of a belt conveyor with variable belt speed based on the material flow rate,“ *Plos One*, January, 2020.
- [18] H. N. P. Z. L. C. B. M. A. Jabbar, „Modelling and Numerical Simulation of a Brushless Permanent – Magnet DC Motor in Dynamic Conditions by Time – Stepping Technique,“ *IEEE Transactions on Industry Applications*, том 40, № 3, 2004.
- [19] A. V. H. M. A. Halvaei Niasar, „Torque Control of Brushless DC Motor Drive based on DSP Technology,“ в *Proceeding of International Conference on Electrical Machines and Systems*, Seoul, Korea, 2007.
- [20] п. И. Петров, *Задвижване и Управление на Модерни Конвейери*, София: Индустириален Софтуер АД, 2017.
- [21] A. Hezam, D. Konstantas и M. Mohammed, „Comprehensive IoT Attacks Survey based on a Building-blocked Reference Mode,“ *International Journal of Advanced Computer Science and Applications(IJACSA)*, том 9, № 3, 2018.
- [22] V. Raja и K. J. Fernandes, *Reverse Engineering: An Industrial Perspective*, London: Springer-Verlag, 2007, p. 3.
- [23] J. E. K.S. Kim, „Expert system for selection of material handling and storage systems,“ *International Journal of Industrial Engineering*, том 4, № 2, pp. 81-89, 1997.

- [24] G. U. T. J. G. Daniel J Fonseca, „A knowledge-based system for conveyor equipment selection,“ *Expert Systems with Applications*, том 26, № 4, pp. 615-623, May 2004.
- [25] „<https://machinelearningmastery.com/lstm-model-architecture-for-rare-event-time-series-forecasting/>“.
- [26] S. Hochreiter и J. Schmidhuber, „LSTM can Solve Hard Long Time Lag Problems,“ *Advances in Neural Information Processing Systems*, том 9, 1997.
- [27] T. C. E. K. Pascal Berruet, „Conveyors With Rollers as Anticipatory Systems: Their Simulation Models,“ в *American Institute of Physics Conference Proceedings*, 2004.
- [28] M. J. P. J. G. J. Berruet P., „Modeling and validation of transitic systems based on reusable components,“ *IIIS/IEEE /SCI 2001*, том 3, № 1, pp. 286-291, July 2001.
- [29] M. P. Fanti, „Event-based controller to avoid deadlock and collisions in zone control AGVS,“ *International Journal of Production Research*, том 40, № 6, pp. 1453-1478, 2002.
- [30] E. Kindler, „Computer Models of Systems Containing Simulating Elements,“ *Computing Anticipatory Systems CASYS 2000 - Fourth International Conference*, том 573, № 1, pp. 390-399, 2001.
- [31] R. A. J. H. P. Roberto Alejo, „An Efficient Over-sampling Approach Based on Mean Square Error Back-propagation for Dealing with the Multi-class Imbalance Problem,“ *Neural Processing Letters*, December 2015.
- [32] W. Kester, „Oversampling Interpolating DACs,“ *Analog Devices*, January 2015.
- [33] A. G. Madhusudan Singh, „Performance Evaluation of BLDC Motor with Conventional PI and Fuzzy Speed Controller,“ *IEEE Transactions on Industrial Electronics*, том 12, № 9, 2012.
- [34] C. S. R. R. M. Surya Kalavathi, „Performance Evaluation of Classical and Fuzzy Logic Control Techniques for Brushless DC Motor Drive,“ *IEEE Transactions on Industrial Electronics*, том 12, № 7, pp. 488-491, 2012.
- [35] J. P. Ryan Helinski, „Measuring Power Distribution System Resistance Variations,“ *IEEE Transactions On Semiconductor Manufacturing*, том 21, № 3, pp. 444-454, 2008.
- [36] S. R. Nassif, „Modeling and analysis of manufacturing variations,“ в *Proc. Conf. Custom Integrated Circuits*, 2001.

- [37] A. Zellner, H. A. Keuzenkamp и M. McAleer, *Simplicity, Inference and Modelling: Keeping it Sophisticatedly Simple*, Cambridge University Press, February 2002.
- [38] T.-H. W. Chang-Lung Tsai, „Information protection from network sniffing of bandwidths sharing,“ *42nd Annual IEEE International Carnahan Conference on Security Technology*, 2008.
- [39] S. Ghosh, A. Basak и S. Bhunia, „How Secure Are Printed Circuit Boards Against Trojan Attacks?,“ *IEEE Design & Test*, August 2014.
- [40] J. B. J. P. H. Ahn, „A Hybrid Non-destructive Measuring Method of Three-dimensional Profile of Through Silicon Vias for Realization of Smart Devices,“ *Scientific Reports*, том volume 8, October, 2018.
- [41] R. W. Hamming, *Error detecting and error correcting codes*, New York: Bell System Technical Publications, 1950.
- [42] E. K. I. T. A. Kindler, „Reflective simulation of logistic and production systems,“ в *International Workshop on Harbour, Maritime and Multimodal Logistics Modelling & Simulation HMS, Modelling & Applied Simulation*, Begeggi, Italy, 2002.
- [43] „Light Load Parts,“ Real Games, Unipessoal Lda, [Онлайн]. Available: <https://docs.factoryio.com/manual/parts/light-load/>.
- [44] M. A. L. L. K. P. T.Na, „Turntable Conveyor,“ Interline System Co.,Ltd., [Онлайн]. Available: <https://www.interlineconveyor.com/album/4535/turntable-conveyor>.
- [45] A.-C. Kasai, „Transfer Modules,“ Itoh Denki Europe S.A.S, [Онлайн]. Available: <https://www.ito-denki.com/en/products/transfer-modules>.
- [46] G. Malina, „Pulse Diverter Unit - PDU90,“ Pulseroller Kyowa EU, 2021. [Онлайн]. Available: <https://www.pulseroller.com/assemblies/pdu90/>.
- [47] A. O. Sudhanshu Mitra, „Dynamic Behavior of Brushless DC Motor under the Application of PI, PD, PID, and Fuzzy Logic Controllers,“ *International Research Journal of Engineering and Technology (IRJET)*, том 2, № 6, p. 916, Sep-2015.
- [48] G. H.-Y. H. I.-S. K. Y. S. Kim. Young-Joo, „Reliability Assessment Criteria of Motorized Roller Conveyor,“ *Journal of The Korean Society of Manufacturing Technology Engineers*, том 24, № 5, 2015.
- [49] S. H. R. S. Richter Michael, „Investigations on the "Direct Digital Inductance Estimation" - Concept for Self-Sensing AMBs Under Influence of Eddy Currents,“ в *Proceedings of ISMB14*, Linz, Austria, Department of

Mechanical Engineering, Institute for Mechatronic Systems in Mechanical Engineering, Oct 2014.

- [50] Д. П. С. С. Г. М. Ивайло Андонов, „Задвижващо средство за ролка от ролков конвейер и метод за приложението му“. България Патент 66930 В1, 4 9 2019.
- [51] К. О. У. М. а. В. Nagai, „Sensorless Magnetic Levitation Control by Measuring the PWM Carrier Frequency Component,“ в *International Symposium on Magnetic Bearings*, 1992.

Декларация за оригиналност на резултатите

Декларирам, че настоящият дисертационен труд съдържа оригинални резултати, получени при проведени от мен научни изследвания (с подкрепата и съдействието на научния). Резултатите, които са получени, описани и/или публикувани от други учени, са надеждно подробно цитирани в биографията.

Настоящата работа не е предлагана за придобиване на научна степен в друго висше училище.

Подпис: