



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

РАЗПРЕДЕЛЕНО УПРАВЛЕНИЕ НА КОНВЕЙЕРНИ СИСТЕМИ

АВТОРЕФЕРАТ

на дисертационен труд за присъждане на образователна
и научна степен
„ДОКТОР“
по научна специалност „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 за важни срокове, 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	4
УВОД	5
I ГЛАВА. ВЪВЕДЕНИЕ	6
I.1 Видове конвейерни системи	6
I.2 Проблеми при захранването на конвейерни системи с МР.....	8
I.3 Проблеми с надеждността на данните в зашумена среда.....	9
I.4 Проблеми свързани със защитата на интелектуалната собственост вложена в модулите за управление	9
I.5 Изкуствен интелект и машинно обучение	9
I.6 Компютърни симулации	10
II ГЛАВА. ДОБАВЯНЕ НА ИНТЕЛИГЕНТНОСТ КЪМ СИСТЕМИТЕ ЗА УПРАВЛЕНИЕ НА КОНВЕЙРИ.....	12
II.1 Подобряване ефективността, чрез разпределено управление	12
II.2 Предложение за подобряване на захранването	Error!
Bookmark not defined.	
II.3 Методи за кодиране на сигналите за позиция на ротора	Error!
Bookmark not defined.	
II.4 Защита на устройствата от клониране.....	21
III ГЛАВА. РЕАЛИЗАЦИЯ НА ПРЕДЛОЖЕНИТЕ РЕШЕНИЯ.....	26
III.1 Подобряване качеството на данните от моторите.....	26
III.2 Подобряване на захранването	28
III.3 Създаване на компютърни симулации	30
IV ГЛАВА. ЕКСПЕРИМЕНТИ И АНАЛИЗ НА РЕЗУЛТАТИТЕ....	32
IV.1 Подобряване качеството на данните	32
IV.2 Експерименти с предложения метод за подобряване на захранването	32
IV.3 Резултати от симулациите	33
IV.4 Експерименти и тестове на цялата система.....	35
IV.5 Заключение.....	35
Приноси	36
Публикации	37

Списък с фигури

Фиг. 1 Централизирано задвижване на еднозонов конвейер	6
Фиг. 2 Разпределено задвижване	7
Фиг. 5 Разположение на обектите спрямо зоните	8
Фиг. 9 Примерна консумирана мощност в течение на времето	8
Фиг. 12 Симулиране на сортировъчен център.....	10
Фиг. 13 Измерване на дължината на зоната	13
Фиг. 14 Всяка зона запазва три списъка с обекти	13
Фиг. 15 Скъсяване на празното разстояние	15
Фиг. 18 Времедиаграма на получените съобщения	17
Фиг. 19 Примерна схема за хранване и комуникация на конвейер .	18
Фиг. 20 Прост пример за разпределяне на хранването	20
Фиг. 24 Използване на половината вътрешна памет за временно място по време на обновяване на софтуера	22
Фиг. 26 Само-обновяване на софтуера от съседните модули.....	23
Фиг. 27 Криптиране на данните	24
Фиг. 28 Изграждане на пръстов отпечатък на устройството, което се използва за криптиране на фърмуера и данните.....	25
Фиг. 32 Влияние на получената грешна информация за положението на ротора върху консумирания ток от мотора	26
Фиг. 33 Софтуер за генерини на комбинациите по Хаминг.....	27
Фиг. 34 Резултат от работа на програмата намиращата равно отдалечени комбинации по Хаминг.....	28
Фиг. 35 Формат на съобщението за консумиран ток.....	28
Фиг. 37 Повишаване на хранващото напрежение при спиране на тежък обект.....	30
Фиг. 38 Код за реализиране на динамично променяне на празното разстояние между обектите	31
Фиг. 39 Задаване на параметри на вградената функционалност за създаване на празно разстояние между обектите	31
Фиг. 59 Осцилограма на напрежението при спиране с използване а новия метод за разпределение на енергията	33
Фиг. 60 Ниска плътност на обектите	33
Фиг. 61 Висока плътност на обектите	33
Фиг. 64 Работа на конвейера без акумулиране на обектите.....	34
Фиг. 65 Акумулация на конвейера след преминаването на 20 обекта	34
Фиг. 66 Експериментален конвейер	35

УВОД

Основна цел на дисертацията е разработването на решение за подобряване ефективността на конвейерни системи, 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 хранване, 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.

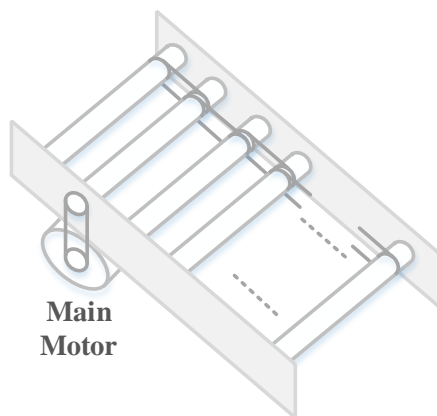
I ГЛАВА. ВЪВЕДЕНИЕ

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.

1.1 Видове конвейерни системи

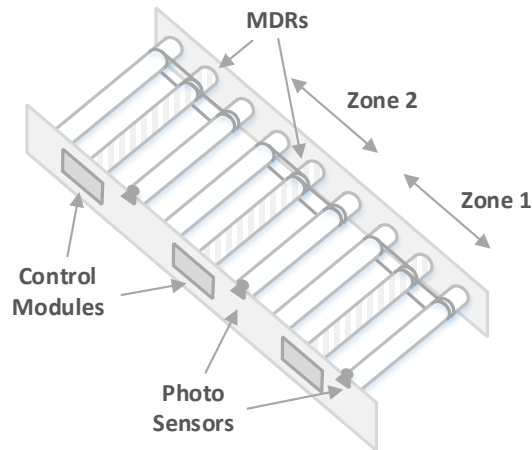
Най-простият вид конвейери представлява група ролки или лента задвижвани едновременно обикновено от електродвигател Фиг. 1. Това представлява една зона за придвижване. Начинът по който обектите са поставени в началото, като ориентация и разстояние по между им остава непроменен до края на придвижването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



Фиг. 1 Централизирано задвижване на еднозонов конвейер

При другият вариант, в който конвейера е образуван от много зони, разстоянието между обектите **може** да се променя по време на придвижването им. Вторият вид задвижване е разпределено от много електродвигатели обикновено вградени в моторизирани задвижващи ролки – (MDRs) Фиг. 2.

Две са основните причини, заради които конвейерите със зони за предпочитани в съоръженията, при които има наличие на работещи хора.. 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. 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.



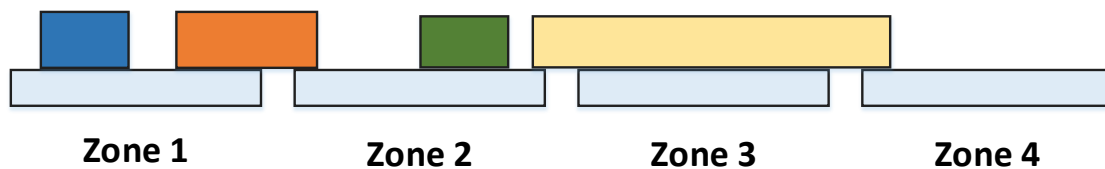
Фиг. 2 Разпределено задвижване

Втората причина е ефективността. 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.

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.

Освен основната цел да придвижи някакво количество обекти от една до друга точка в пространството от конвейерната систем се очакват още функции, като буфериране на обектите, промяна на ориентацията (вертикалното завъртане), дори пренареждане на подредбата на обектите.

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. 3.

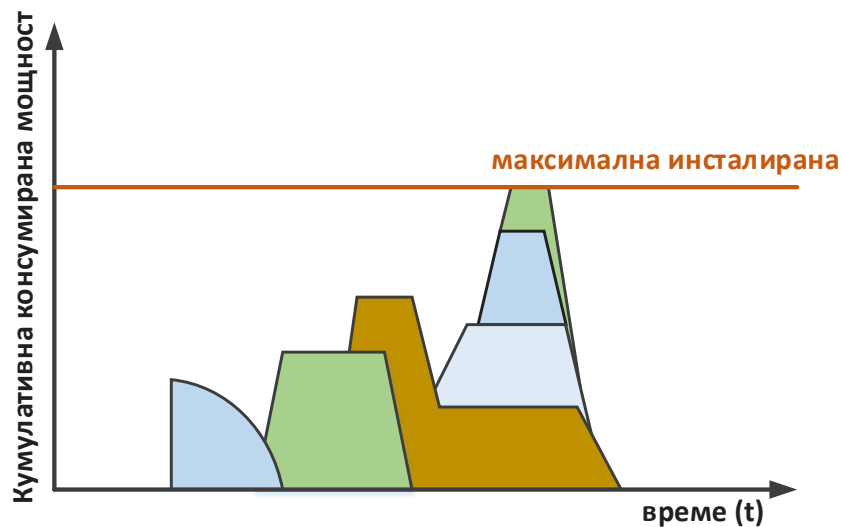


Фиг. 3 Разположение на обектите спрямо зоните

1.2 Проблеми при захранването на конвейерни системи с МР

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.

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.



Фиг. 4 Примерна консумирана мощност в течение на времето

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 връщат към мрежата или да я разсейват.

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

1.3 Проблеми с надеждността на данните в зашумена среда

При комутацията на БЧПТ двигатели се използва информацията от сензори за положението на ротора на мотора.

Предаването на данните може да е директно, като сигнали от датчиците по отделни проводници или кодирано по един проводник. 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.

1.4 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.

Компанията разработила контролера обикновено има многократно по-големи инвестиции за разработването на софтуера, в сравнение с хардуера. От друга страна за конкурентна компания може да се окаже по-малка инвестиция обратното инженерство и дублирането на хардуера, както и прочитането на софтуера от готов контролер, отколкото да създава двете на ново.

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.

1.5 Изкуствен интелект и машинно обучение

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

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.

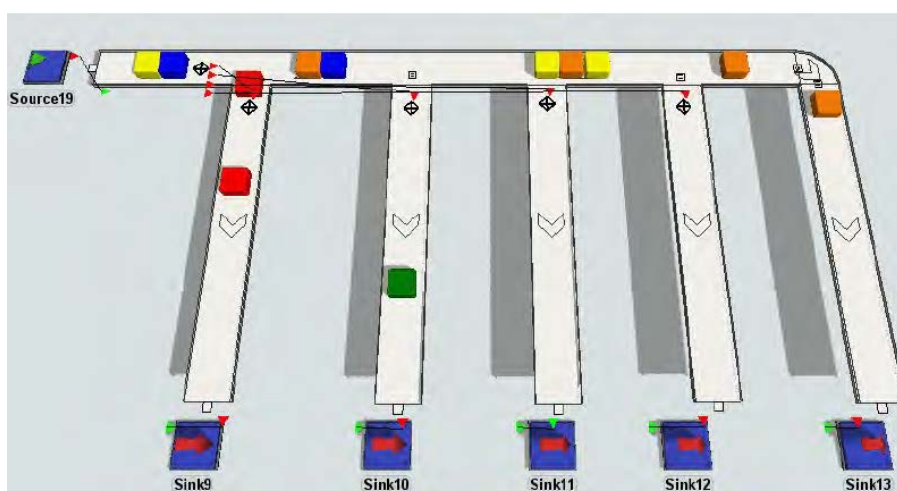
Изкуственият интелект е свързан с проявата на интелигентно поведение. За разлика от човешкия интелект обаче, изкуственият се осъществява в неща, които са създадени изкуствено, като например компютърни програми.

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.

1.6 Компютърни симулации

Подходящи за компютърни симулации са софтуерни продукти работещи в дискретно време, базирани на събития от симулирания процес, като FlexSim и SimCAD и др. Първият е по-удобен, заради приятелския интерфейс и лесния начин за създаване на модели, Фиг. 5.

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



Фиг. 5 Симулиране на сортировъчен център.

1.7 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 надежността при предаване на данните между електрическите мотори и контролера.
4. Да се предложи и имплементира метод за подsigуряване на контролерите срещу клониране.

II ГЛАВА. 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 Подобряване ефективността, чрез 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.

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.

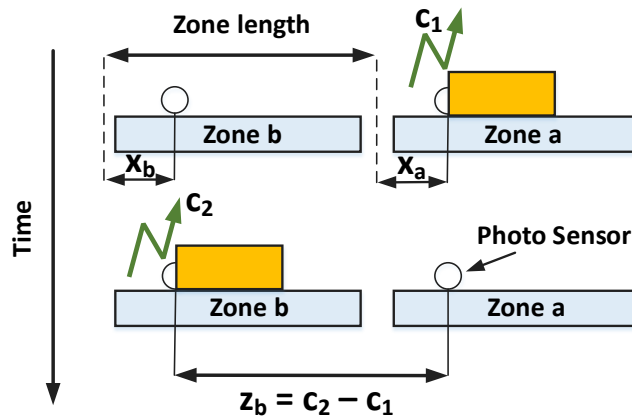
Дадена зона може при получаване на обект от предхождащата зона да запомни стойността на това разстояние като c_1 . Когато на сензорът ѝ се появи сигнал тя записва стойността на това разстояние като c_2 . Има и някакво разстояние между фотосензора на предхождащата зона и края ѝ, което ще бъде отбелязано 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. 6. 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):

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

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



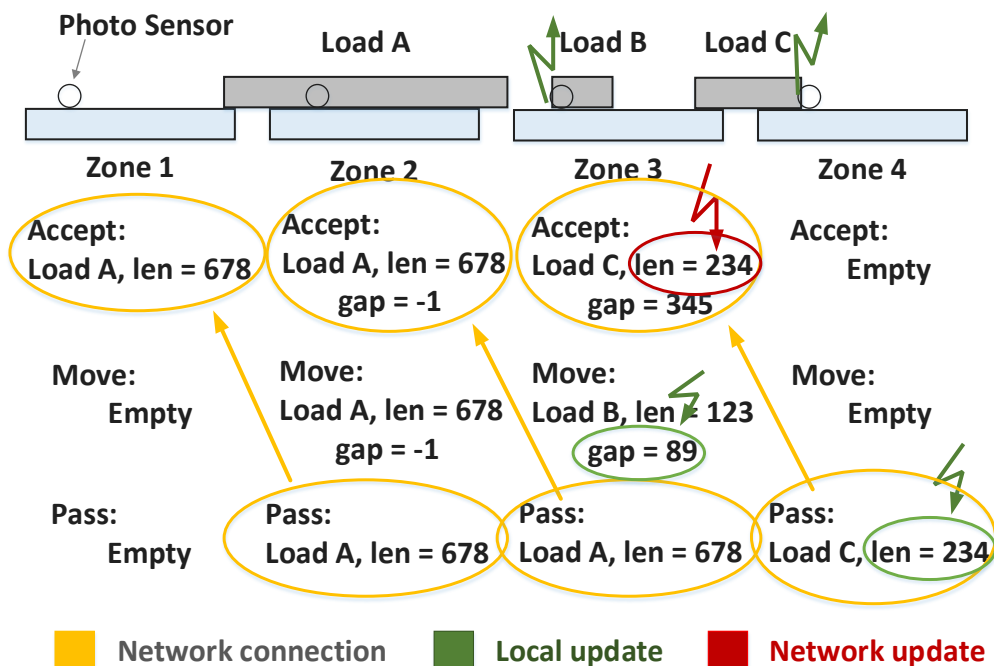
Фиг. 6 Измерване на дължината на зоната

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 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. 7 Всяка зона запазваща три списъка с обекти

На Фиг. 7 е показано предложение, в което всяка зона поддържа три списъка с обекти:

- Които се приемат в момента с дължина 0 или 1 наречен Assert.
- 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. 7 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.

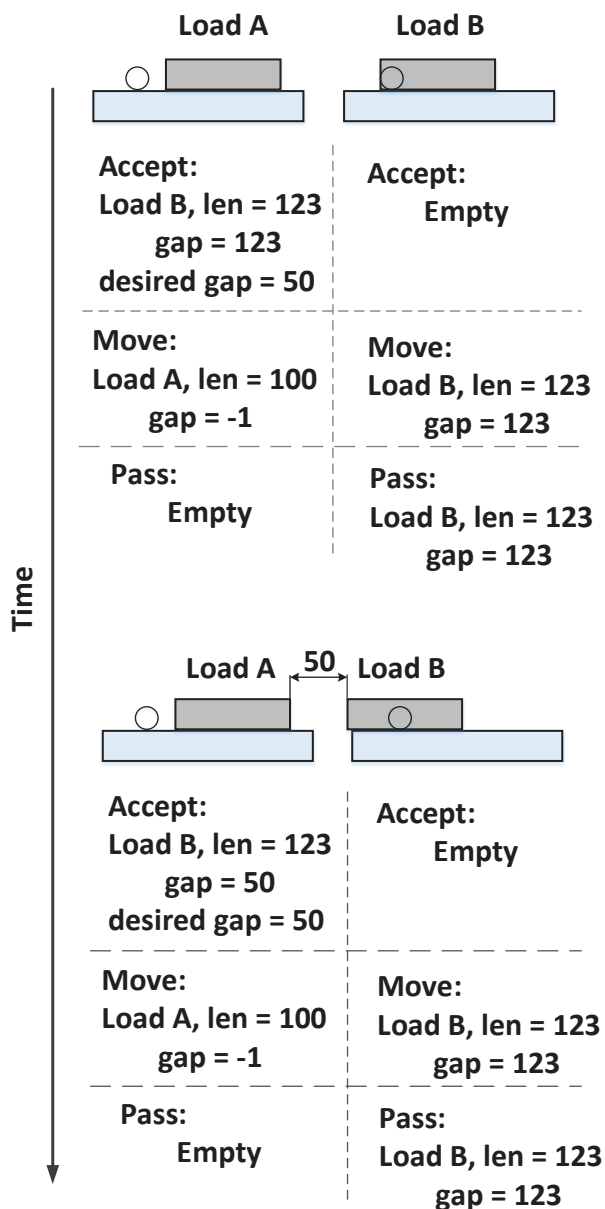
Първата и последната зони могат да се определили като такива използвайки информация за разликата във функционалността на съседните зони, предавана по мрежата. Използвайки тази информация всяка зона между тях може да определи относителното си положение изразено като разстояние спрямо началото и края на линейния участък.

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.

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. 8. 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. 8 Скъсяване на празното разстояние

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.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.

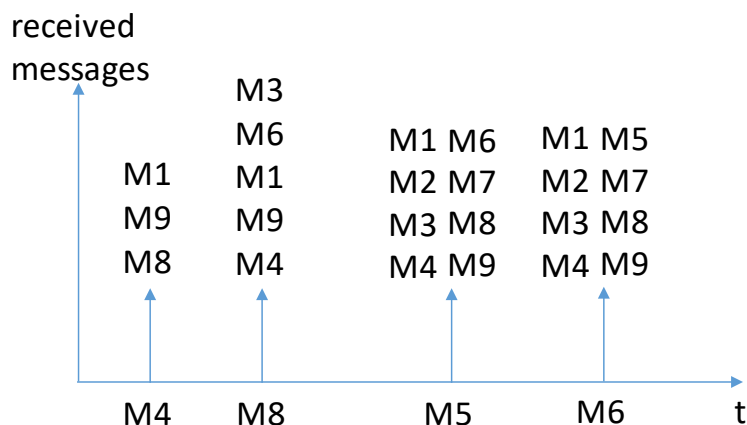
Обикновено 100% от върнатата енергия спиране на моторизираните ролери се преобразува в топлина вътре в контролерите за управление. С предложени метод цялата или част от нея може да бъде използвана и така да се постигне много по-добра енергийна ефективност.

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.

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

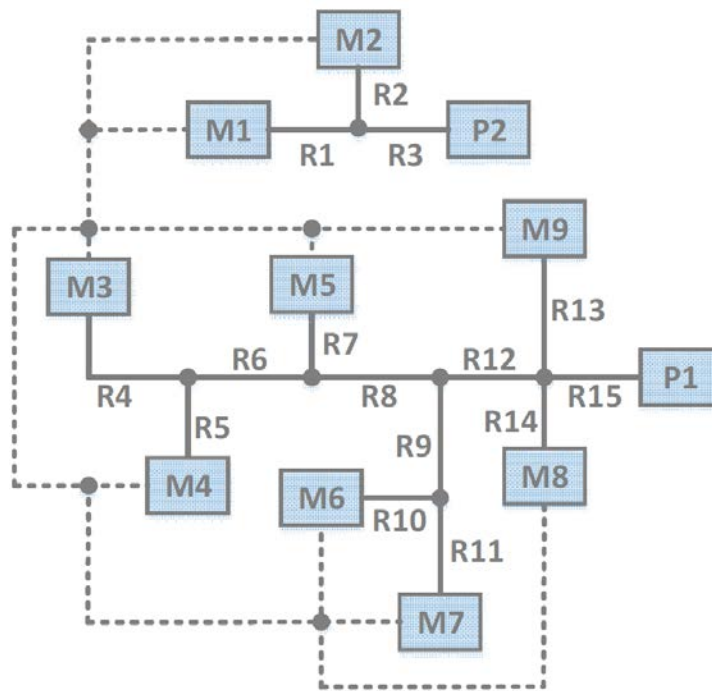


Фиг. 9 Времедиаграма на получените съобщения

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 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. 9. 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. 10 Примерна схема за захранване и комуникация на конвейер

Нека разгледаме примерна конвейерна система показана на Фиг. 10. Прекъснатата линия показва комуникационния канал, а непрекъснатата захранващата линия. Блоковете с М са модули за управление на зона(и) на конвейер, а модулите с Р са захранващи блокове. С R са отбелязани съпротивленията на участъците от захранващия кабел.

Видимите един за друг модули, които има обща мрежа на комуникация и имачи 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 модулет 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 захранващи 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 захранващото 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 захранващ блок.** Ако V_{0m_y} и V_{Lm_y} измерени от модул m_y са равни това означава, че захранващият 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 захранващия 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 захранването, 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 предишния и означава, че няма смисъл да се прави извод в този случай, ако в предишния е установено, че m_x и m_y са захранени от различни блокове. Модул m_y изчислява съпротивлението между точката на разклонение S_{xy} и захранващия блок включвайки вътрешното съпротивление на захранващия блок по (9):

$$R = \frac{V_0 - V_L}{I_L} \quad (9)$$

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 захранващия 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 захранващият блок и модул m_x . В този случай модул m_y може да изчисли съпротивлението между него и захранващия блок включвайки вътрешното съпротивление на захранващия блок R_1 по (10).

$$R_1 = \frac{V_0 - V_{Lm_y}}{I_L} \quad (10)$$

Както и съпротивлението между възела, от който е разклонено неговото захранване S_{yx} и модул m_x съответно R_2 по (11).

$$R_2 = \frac{V_0 - V_{Lm_y} + V_{Lm_x}}{I_L} \quad (11)$$

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 захранващата линия също извършва изчисление на общото съпротивление между него и захранващия блок включвайки вътрешното му съпротивление по (9).

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.

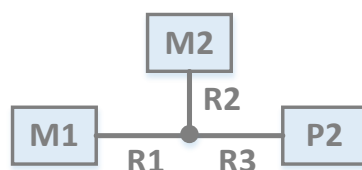
За простота нека разгледаме системата показана на Фиг. 11. В този случай ще има два списъка с еднакво име:

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. 11 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 модулет, 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 модулет намери, че е свързан някъде в участъка на пътя между модула изпратил записа и захранващия блок или обратното, че модулет изпратил записа е свързан някъде по пътя между модула получил записа и захранващия блок се прави сравнение на стойността на съпротивлението на участъка между захранващия източник и точката на разклонение като се използва малък хистерезис толериращ съвпадение. 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.

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

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.

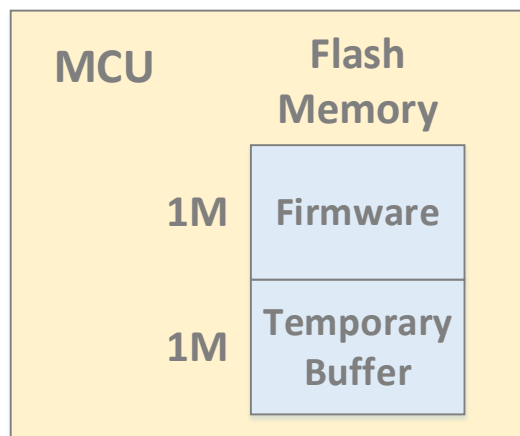
- Метод за кодиране на данните чрез равно отдалечени по Hamming комбинации.
- Метод за предсказване на най-вероятното следващо състояние. Състоянието се предава 4 пъти по 3 бита. В зависимост от очакваното състояние по посока на въртенето и наличието на това състояние в поне 2 места, то го приемаме за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

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.

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.

Друг случай, в който се налага използването на шини за комуникация е когато не може да се реализира цялото управление в един микроконтролер.



Фиг. 12 Използване на половината вътрешна памет за временно място по време на обновяване на софтуера

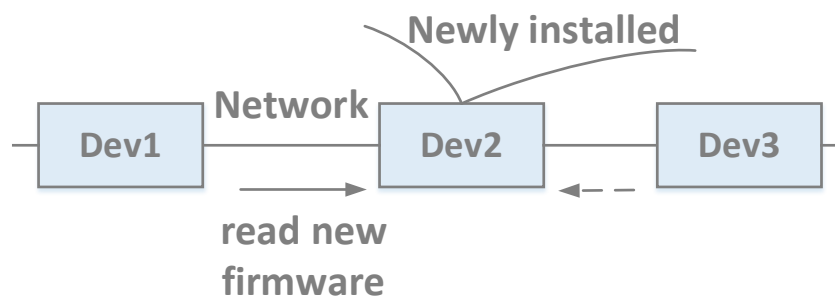
Външните шини могат да бъдат подслушвани и ако данните по тях не са криптирани те могат да бъдат записани синхронно заедно с поведението на контролера, и така да бъде разгадано тяхното значение. И докато разгадаването на значението на данните може да не е чак толкова голяма беда, по някои шини може да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

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 произведени от един производител има заложені 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. 13.



Фиг. 13 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

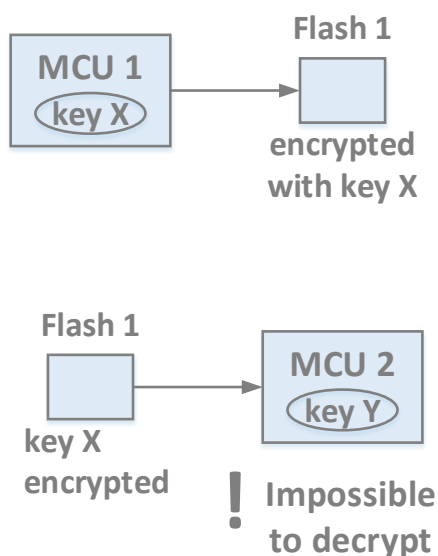
Всички видими части на устройството, като кутия, разположение на компонентите и т.н. е невъзможно да бъдат скрити и затова е лесно да бъдат копирани.

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.

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.

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.

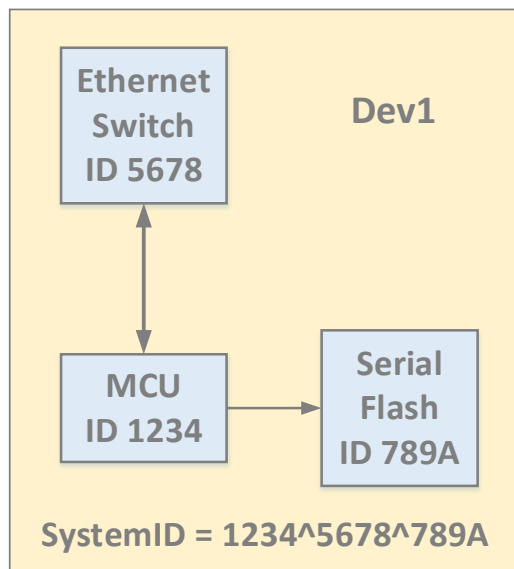


Фиг. 14 Криптиране на данните

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. 14.

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 който 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. 15 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.

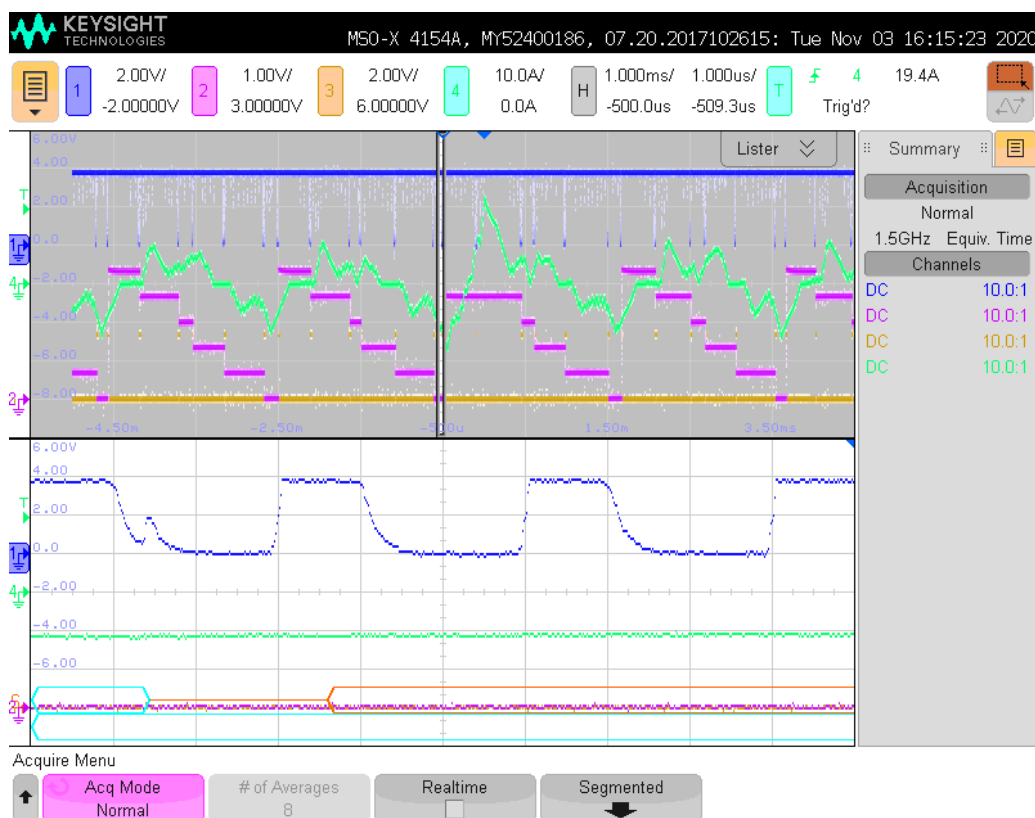
В случай, че за комуникацията с външния свят се използва симетрично криптиране, е добре към ключа да се добави някаква стойност, известна и за двете страни (IoT устройството и компютъра), като например “TCP window sequence” номера, MAC адреси и т.н. Това ще направи шифрования файл винаги различен и за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

III ГЛАВА. РЕАЛИЗАЦИЯ НА ПРЕДЛОЖЕНИТЕ РЕШЕНИЯ

III.1 Подобряване качеството на данните от 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:

На Фиг. 16 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. 16 Влияние на получената грешна информация за положението на ротора върху консумирания ток от мотора

Разработен е софтуер, който да намери равно отдалечени по Хаминг 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. 17 и Фиг. 18.

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.

```
        /* 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> lbGetBytesOnHammingDist(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. 17 Софтуер за генерирне на 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. 18 Резултат от работа на програмата намиращата равно отдалечени комбинации по Хаминг

III.2 Подобриване 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. 19 Формат на съобщението за консумиран ток

В съобщението показано на Фиг. 19 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 0,5A. 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.

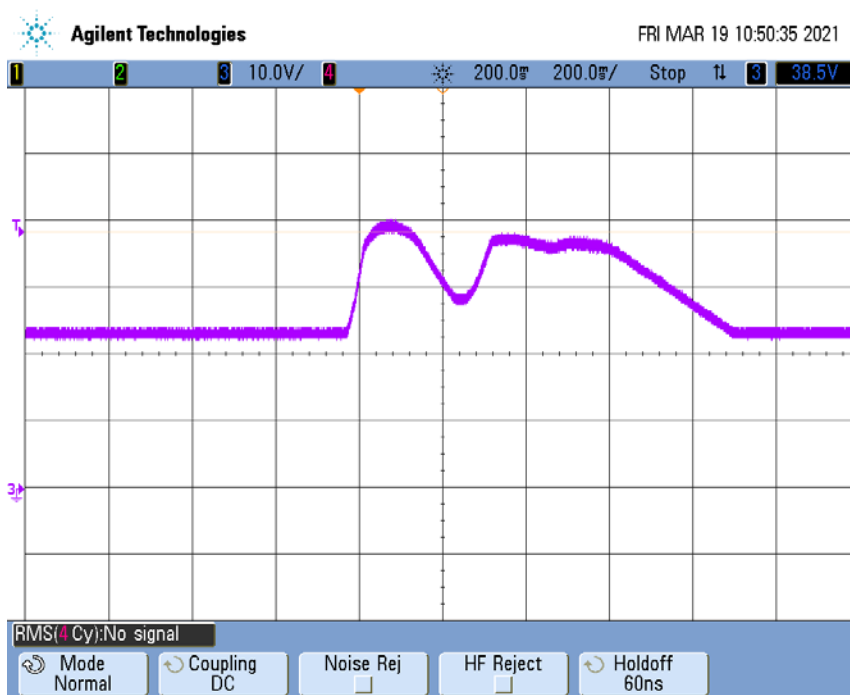
Модулът, който връща енергията, изчислява на спада на напрежението, започвайки от захранването, докато включва максимално допустимото предварително зададено напрежение.

Продължавайки надолу по линията и използвайки съпротивленията 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.

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 и Фиг. 20 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. 20 Повишаване на захранващото напрежение при спиране на тежък обект.

III.3 Създаване на 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. Създаден е код, който реализира

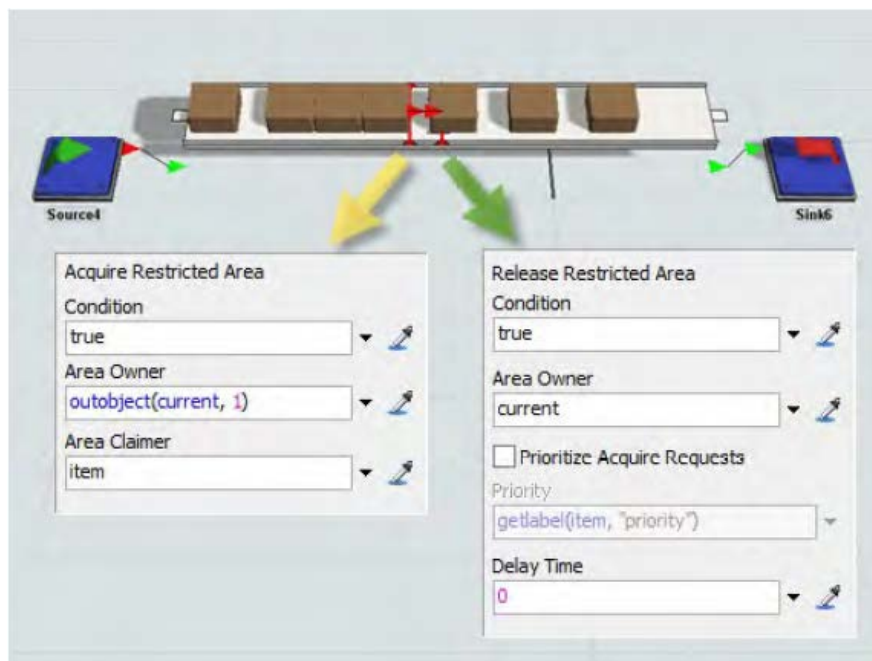
идеята за динамично променяне на празното разстояние между обектите, чийто код е показан на Фиг. 21.

```

Conveyor current = ownerobject(c);
Object item = param(1);
{ // Cv2
    Object involved = item;
    //current.targetSpeed = current.currentSpeed * 0.7;
    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;
}

```

Фиг. 21 Код за реализиране на динамично променяне на празното разстояние между обектите



Фиг. 22 Задаване на параметри на вградената функционалност за създаване на празно разстояние между обектите

Софтуерът има някои вградени функционалности, Фиг. 22, които могат да бъдат използвани само с настройването на параметри в някои менюта.

IV ГЛАВА. ЕКСПЕРИМЕНТИ И АНАЛИЗ НА РЕЗУЛТАТИТЕ

IV.1 Подобряване качеството на данните

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]
```

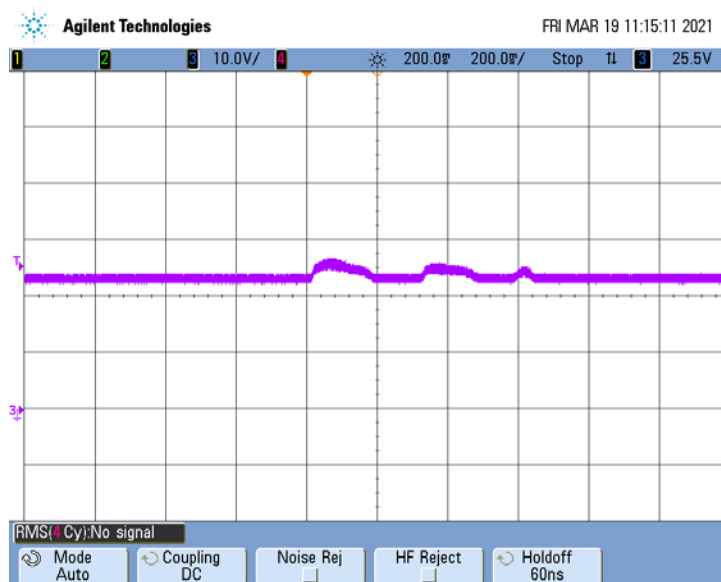
```
MotMain:[81820636]Old(hall:2, pos:5), ExpHall:2,6,3, New(hall:2, pos:5), RX:[10|2|2][00|4|4]<-[10|2|2][10|2|2]<-[10|2|2][10|2|2]
```

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. 20 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, който 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. 23 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. 23 Осцилограма на напрежението при спиране с използване а новия метод за разпределение на енергията

IV.3 Резултати от симулациите

Конвейерът, използващ предложеният алгоритъм създава годими празни пространства между товарите 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. 24 Ниска плътност на обектите

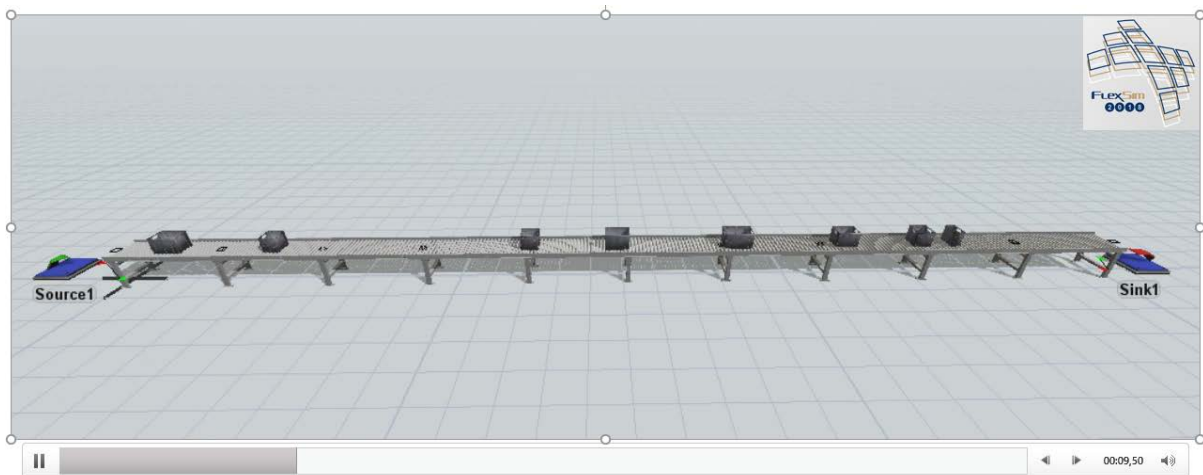


Фиг. 25 Висока плътност на обектите

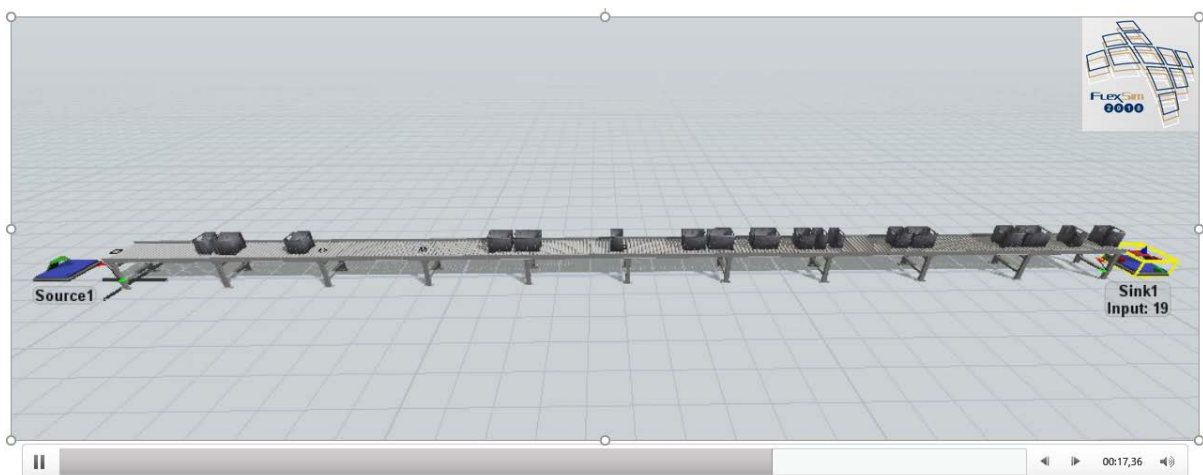
Зоните могат да продължат измерването на размерите на натоварите и празнините и да правят статистическо усредняване на стойността, с цел да я доближат до реалната. Заедно със стойността за всеки размер (на товара или празнината), зоните могат да обменят допълнителна информация, показваща колко вероятно е тази стойност да представлява реалния размер.

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. 26 Работа на конвейера без акумулиране на обектите



Фиг. 27 Акумулация на конвейера след преминаването на 20 обекта

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. 28.



Фиг. 28 Експериментален конвейер

Компютърните симулации, както и проведените експерименти дават обнадеждаващи резултати.

IV.5 Заключение

Благодарение на предложени иновативен метод за разпределено управление на конвейерни системи се 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.

Предложеният метод за подсигуряване 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 нови методи за разпределено управление;
- Предложени и имплементирани са нови методи подобряващи работата на подсистемите за измерване, хранване 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 предложените решения;
- 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

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: