

СИСТЕМА ЗА ОТЧИТАНЕ ПРОИЗВОДИТЕЛНОСТТА НА ТРУДА В РЕАЛНО ВРЕМЕ

Станимир Йорданов¹, Стефан Иванов², Георги Михалев³
^{1 2 3} Технически Университет Габрово

SYSTEM FOR AUTOMATIC REPORTING A LABOUR PRODUCTIVITY IN REAL TIME

S. Yordanov¹, S. Ivanov², G. Mihalev³
^{1 2 3} Technical University of Gabrovo,

Abstract

In this paper, an ESP32 application in implementing systems for automated reporting of the utilization rate of production capacities in real time is considered. A technical solution of remote control and monitoring of the production process in a textile company, through a dedicated module for sending the time spent on a remote server via Wi-Fi, Lan or Bluetooth is proposed. The obtained information is used to control the production process in real time and determination of the remuneration of workers.

Keywords: Internet of Things (IoT), Wi-Fi module ESP32, Arduino IDE development environment, LPWAN technologies, electronic accounting technologies

ВЪВЕДЕНИЕ

Повишаването на производителността на труда е основен фактор за подобряване на икономическите показатели, финансовото състояние и ефективността на производствено-стопанската дейност на фирмата. От друга страна производителността на труда е съставен икономически показател, който съпоставя постигнатия резултат (новосъздаден продукт) с вложения трудов фактор при осъществяването на определена икономическа активност на определена икономическа територия за определен период [1]. Един от компонентите му е оптималното използване на машините и съоръженията в процеса на производство.

Използването на производствената мощност, както и използването на дълготрайните материални активи се характеризира основно с три показателя [2]:

- **Коефициент за екстензивно използване или използване на производствената мощност по време.** Той представлява отношението на фактиче-

ския фонд отработено време към ефективния фонд работно време;

- **Коефициент за интензивно използване на производствената мощност.** Той характеризира използването на производствената мощност в единица време и се определя като отношение на фактически изпълнената работа за единица отработено време към проектната техническа норма;
- **Показатели за интегрално използване.** Характеризират най-пълно използването на производствените мощности. Определят се като произведение на коефициента за екстензивно използване и коефициента на интензивно използване.

Особен интерес за менажерите на предприятията представлява коефициента за интензивно използване на производствените мощности. Той носи информация както за реалното натоварване на производствените ресурси, така и за състоянието на технологичното оборудване, трудовата дисциплина, материално-техническата снабденост на

предприятието и не на последно място квалификацията на обслужващия персонал. Настоящата разработка е следствие на техническо задание възложено от текстилна фирма. Статията е продължение на [6] и описва сървърната част на разработката.

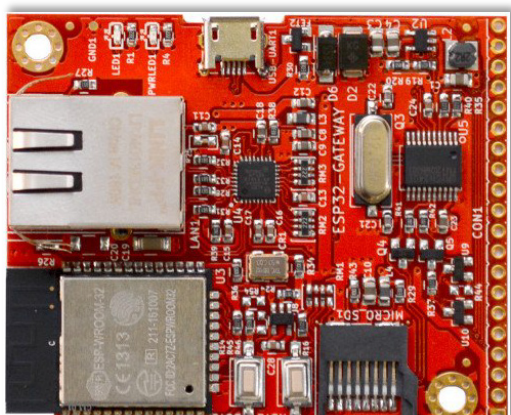
ИЗЛОЖЕНИЕ

Накратко системата за мониторинг е разработката на базата на техническо задание със следните основни пункта:

- Разработената на автономна интегрирана система отчитаща работното състояние на текстилни машини;
- Информация за състоянието на машините се получава от индуктивни сензори;
- Системата да има възможност за разширение;
- Връзката на контролерите с базата от данни да бъде през Ethernet или WiFi;
- При отпадане на захранването или връзката със сървъра, данните от измерванията да се съхраняват и изпратят до сървъра в подходящ момент;
- Наличие на специализиран софтуер за обработка и визуализация на получените входни данни от всяка една текстилна машина и др.

За реализирането на заданието е стъпено на основата на мрежово свързано смарт устройство чиято архитектура и функционалност е описана в [6].

Смарт контролера е изграден на базата на развойната система ESP32-GATEWAY development board with WiFi BLE Ethernet, micro SD card UEXT and GPIO производство на фирмата OLIMEX Ltd. Пловдив (фиг. 1).



Фиг. 1. Развойна система Olimex ESP32-GATEWAY

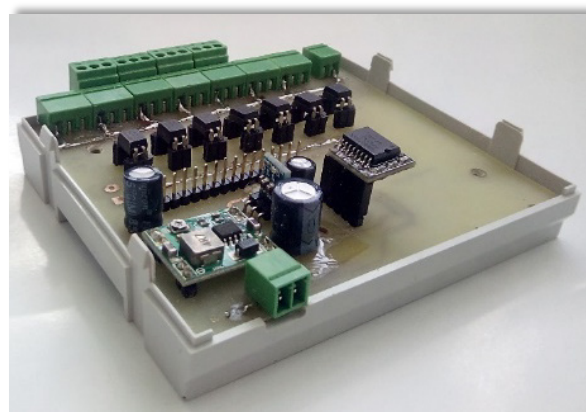
Тя е изградена на базата на микроконтролера ESP-WROOM-32 и притежава Ethernet 100Mb интерфейс, Micro USB конектор за връзка по UART с контролера, Micro SD карта.

Този тип системи намират приложение при експериментиране и изграждане на Internet of Things (IoT) [3,5], проекти и прототипи. В софтуерно отношение платформата е силно развита. Освен основният системен софтуер на Espressif, който е AT базиран (сериен с AT команди) са налични други методи за програмиране. На ниско ниво SDK на Espressif - в два варианта Non OS и Free RTOS. За любителите на LUA NodeMcu. В настоящата разработка за програмирането на Wi-Fi модула въз основа на чипа ESP32 е използвана средата за разработване Arduino IDE [4]. и тулчейна Arduino ESP master, който може да бъде свален от сайта

<https://github.com/espressif/arduino-esp32>.

Ардуино тулчена е изграден върху Espressif IoT Development Framework.

За реализиране на връзката със сензорите е разработена допълнителна разширителна интерфейсната платка показана на (фиг. 2).



Фиг. 2. Разширителна интерфейсната платка

Функциониране на системата

Разработената система е изградена на три нива: Ниво контролери, Сървърно ниво и конфигуриращ софтуер.

Първото ниво представлява мрежа от смарт контролери събиращи информация за отработеното време на машините. На всяка минута контролерите изпращат информация за отработеното време от машините към базата с данни разположена на отдалечен сървър. Връзката със сървъра се осъще-

ствява по Ethernet и WiFi мрежа, като е възможна едновременната работа на двете мрежи. Ако са включени и двете мрежи, данните до базата данни се предават по сигурната жична линия Ethernet. В тази ситуация WiFi мрежата се използва за параметриране на контролера и за следене състоянието на процесите свързани към контролера чрез локална WEB страница (Фиг.3.). В тази страница се визуализира информацията за работата на машините за последните 30 минути.

deviceID	nDate	nTime	Sens0	Sens1	Sens2	Sens3	Sens4	Sens5	Sens6	Sens7
100003	2018-10-05	10:15:04	0	60	0	60	0	0	0	0
100003	2018-10-05	10:14:04	0	60	60	60	0	0	0	0
100003	2018-10-05	10:13:04	0	60	40	60	0	0	0	0
100003	2018-10-05	10:12:04	0	60	60	60	0	0	0	0
100003	2018-10-05	10:11:04	0	60	60	60	0	0	0	0
100003	2018-10-05	10:10:04	0	60	60	60	0	0	0	0
100003	2018-10-05	10:09:04	0	60	40	60	0	0	0	0
100003	2018-10-05	10:08:04	0	60	60	60	0	0	0	0
100003	2018-10-05	10:07:04	0	60	60	60	0	0	0	0
100003	2018-10-05	10:06:04	0	60	60	60	0	0	0	0
100003	2018-10-05	10:05:04	0	60	60	60	0	0	0	0
100003	2018-10-05	10:04:04	0	60	20	60	0	0	0	0
100003	2018-10-05	10:03:04	0	60	40	0	0	0	0	0
100003	2018-10-05	10:02:04	0	60	60	0	0	0	0	0
100003	2018-10-05	10:01:04	0	60	60	0	0	0	0	0
100003	2018-10-05	10:00:04	0	60	60	60	0	0	0	0

Фиг. 3. Web приложение

Времето за визуализация на данните съобразено с буфера на цикличната опашка отговаряща за изпращане на данните към сървъра. Параметрирането на контролерите се извършва през UART или UDP посредством командни низове от вида !WIP:192.168.1.152. На всяка минута данните от измерванията се съхраняват във файлове разположени на microSD карта памет и се изпращат към базата с данни на сървъра. Данните от файлове разположени на SD картата се достъпват автоматично от SQL сървъра посредством съхранена процедура или постъпили запитвания по UDP или UART, или през FTP сървър изграден на контролера.

Второто ниво (Система за управление на базата с данни) на системата е разположено на отдалечен сървър. На него е инсталирана релационна база от данни в която се

съхраняват данните от измерванията и данните за конфигуриране на контролерите.

Използвана е релационна база от данни MySQL 5.3. Посредством SQL заявки контролерите добавят в таблица measures информация за седем машини свързани към определен контролер. Добавянето на запис инициран от контролерите в базата задейства тригери и съхранени процедури, които оценяват данните и попълват автоматично таблици с информация за работата по смените на отделните машини.

Тригерът задействан при постъпване на данните актуализира таблица measuresHour.

Всеки един запис от таблицата съдържа информация за отработеното време за определен час от машините подключени към контролер с идентификационен номер записан в полето deviceID. Тригерът вмъква нов запис в началото на всеки час или прави update на полетата от таблицата measuresHour при всяко добавяне на запис в таблица measure. Ето и едно примерно решение на тригера:

```
DELIMITER $$
drop TRIGGER if EXISTS measures_INSERT $$
CREATE TRIGGER `measures_INSERT`
AFTER INSERT ON `measures` FOR EACH ROW
BEGIN
    if minute(new.nTime) = 0
    or (not EXISTS (SELECT 1 FROM
measuresHour WHERE nDate = new.nDate and
nTime = MAKETIME(HOUR(new.nTime),0,0)))
    then
        INSERT INTO measuresHour ( deviceID,
nDate, nTime, Sens0, Sens1)
values ( new.deviceID, new.nDate,
MAKETIME(HOUR(new.nTime),0,0),
new.Sens0, new.Sens1);
    ELSE
        UPDATE measuresHour
SET
Sens0 = Sens0 + new.Sens0 ,
Sens1 = Sens1 + new.Sens1
WHERE deviceID = new.deviceID AND
nDate = new.nDate and nTime =
MAKETIME(HOUR(new.nTime),0,0);
    END IF;
END $$
DELIMITER ;
```

При отсъствие на връзка с даден контролер за период по-голям от 5 минути се генерира заявка и се изпраща e-mail до определена група лица описани в таблица users. Проверката на връзката се извършва от събитие изпълнявано на всяка минута. Структурата на едно подобно събитие е следната:

```

DELIMITER $$
SET GLOBAL event_scheduler = ON $$
DROP EVENT IF EXISTS eventCheckConnect $$
CREATE EVENT eventCheckConnect
ON SCHEDULE EVERY 1 minute
STARTS '2018-04-09 04:30:00'
DO begin
CALL checkConnect ( );
END $$
DELIMITER ;

```

Имейла се генерира от PHP скрипт за-действан от съхранена процедура. Кодът на задействане на PHP скрипта е следният:

```

SET cmd = CONCAT('php.exe -f
"/sendMail.php"', ' ');
SET result = sys_eval(cmd);

```

На определен период от време автоматично се генерират дневни, седмични и месечни отчети в Excel формат. Тези отчети се изпращат чрез e-mail към оторизираните лица от PHP скрипт активиран от базата данни. Месечния отчет се генерира веднъж на първо число, а седмичния всеки понеделник. На фигура 4 е показан един седмичен отчет.

Справка за КИМ										
2018-09-24 - 2018-09-30										
Дата	Име на машината	Смяна 1 КИМ [%]	Смяна 1 Раб. часове	Смяна 2 КИМ [%]	Смяна 2 Раб. часове	Смяна 3 КИМ [%]	Смяна 3 Раб. часове	Редовна смяна КИМ [%]	Редовна смяна Раб. часове	
24.09.2018	Карит към Двора	85	07:13:30	80.098	06:48:30	0	00:00:00	0	00:00:00	
25.09.2018	Карит към Двора	84.4771	07:10:50	73.2026	06:13:20	58.254	04:04:40	0	00:00:00	
26.09.2018	Карит към Двора	73.9869	06:17:20	66.5033	05:39:10	76.7063	05:22:10	0	00:00:00	
27.09.2018	Карит към Двора	73.1046	06:12:50	67.451	05:44:00	71.1111	04:58:40	0	00:00:00	
28.09.2018	Карит към Двора	75.2941	06:24:00	70.7516	06:00:50	66.4286	04:39:00	0	00:00:00	
29.09.2018	Карит към Двора	0	00:00:00	0	00:00:00	0	00:00:00	0	00:00:00	
30.09.2018	Карит към Двора	0	00:00:00	0	00:00:00	0	00:00:00	0	00:00:00	
Общо за периода		78.3725	33:18:30	71.6013	30:25:50	68.125	19:04:30	0	00:00:00	
24.09.2018	Първа Линия	22.6797	01:55:40	0	00:00:00	0	00:00:00	0	00:00:00	
25.09.2018	Първа Линия	41.4379	03:31:20	57.1895	04:51:40	72.4603	05:04:20	0	00:00:00	
26.09.2018	Първа Линия	83.0065	07:03:20	76.7974	06:31:40	75.3968	05:16:40	0	00:00:00	
27.09.2018	Първа Линия	57.451	04:53:00	85.2288	07:14:40	83.254	05:49:40	0	00:00:00	
28.09.2018	Първа Линия	73.5294	06:15:00	82.6797	07:01:40	56.0317	03:55:20	0	00:00:00	
29.09.2018	Първа Линия	86.2092	07:19:40	18.6275	01:35:00	0	00:00:00	0	00:00:00	
30.09.2018	Първа Линия	0	00:00:00	0	00:00:00	0	00:00:00	0	00:00:00	
Общо за периода		60.719	30:58:00	64.1046	27:14:40	71.7857	20:06:00	0	00:00:00	
24.09.2018	Втора Линия	74.8366	06:21:40	0	00:00:00	0	00:00:00	0	00:00:00	

Фиг. 4. Седмичен отчет.

Отчета съдържа отработеното време по смени за всеки ден, както и коефициента за използване на машината (КИМ) за всеки ден от седмицата. На края се прави обобщение за периода. Коефициента за използване на машината се изчислява без да се вземат в предвид регулярните почивки.

PHP скриптът отговорен за изпращане на e-mail съобщения с отчетите до оторизира-

ните лица има приблизително следният КОД:

```

function sendReportMail() {
if (!$this->DBLogin()) {
$this->HandleError("DB login failed!");
return false;
}
$sqlMails = "select * from Reports..";
$retval =
$this->connection->query( $sqlMails );
if ($row = $retval->fetch(
PDO::FETCH_ASSOC)) {
$mailer = new PHPMailer();
$mailer->CharSet = 'utf-8';
$mailer->isSMTP();
$mailer->SMTPDebug = 0;
$mailer->Debugoutput = 'html';
$mailer->Host = 'smtp.abv.bg';
$mailer->Port = 465 ;
$mailer->SMTPSecure = 'ssl';
$mailer->SMTPAuth = true;
$mailer->Username = "sys@abv.bg";
$mailer->Password = "12345";
$mailer->setFrom(sys@abv.bg);
$mailer->addReplyTo('sys@abv.bg','');
$mailer->From = 'sys@abv.bg';
$mailer->Subject = $mSubject;
$mailer->Body = "Здравейте г-не\n
На вашето внимание предоставяме\n
.$row['mBody'] ."\n генериран на "
.$row['added_date']
." \n\n С уважение, \n\t\t admin\n";
$mailer->addAttachment('Report.xls');
if (!$mailer->Send()) {
$this->HandleError("Err send email");
return false;
}
unset($mailer);
return true;
}
}

```

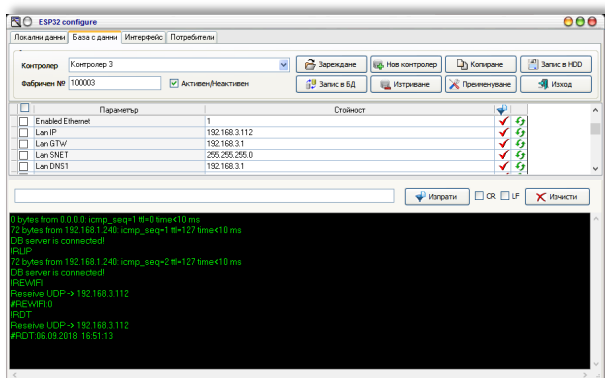
Информацията за отработеното време се съхранява в динамични таблици (view) генерирана от следния скрипт.

```

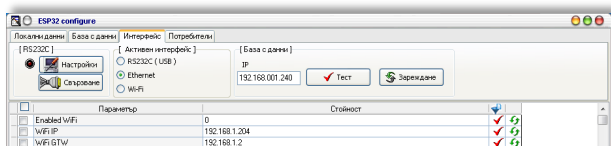
delimiter $$
drop view IF EXISTS shiftTimeView $$
create view shiftTimeView as
select hw.deviceID, hw.sensorID,
dv.Titles as devTitle, sn.Titles as
sensTitle, hw.nDate ,
SEC_TO_TIME(hw.WorkShift_0) as
shiftTime_0, hw.WorkShift_0/288.0 as KP0,
SEC_TO_TIME(hw.WorkShift_1) as
shiftTime_1, hw.WorkShift_1/288.0 as KP1,
SEC_TO_TIME(hw.WorkShift_2) as
shiftTime_2, hw.WorkShift_2/288.0 as KP2,
SEC_TO_TIME(hw.WorkShift_3) as
shiftTime_3, hw.WorkShift_3/288.0 as KP3
from hoursWorked_new hw, devices dv,
sensors sn
where hw.deviceID = dv.deviceID and
dv.Enabled = true and sn.deviceID =
dv.deviceID and sn.sensType > 0 and
hw.sensorID = sn.sensorID;
$$
delimiter ;

```

Освен информацията за отработеното време в базата с данни се съхранява и информация за работните графици по машини, информация за настройка на контролерите и сензорите и др.. За управлението на тази информация се грижи „Софтуер за конфигуриране и диагностика“ (фиг. 5) представляващ третото ниво от системата. Разработен е в средата C++ Builder 6.0 като 32 битово приложение с възможност за работа под всички версии на Windows след Windows 98. Програмата комуникира с контролера по интерфейсите RS232 и Ethernet, а със сървъра посредством ODBC драйвери за връзка с MySQL (фиг. 6).



Фиг. 5. Софтуер за конфигуриране и диагностика



Фиг. 6. Избор на протокол за връзка с контролерите

Командите са текстови със следният формат:

!WIP:192.168.1.152

При правилно приета команда контролера отговаря с телеграма във формат:

#WIP:192.168.1.152

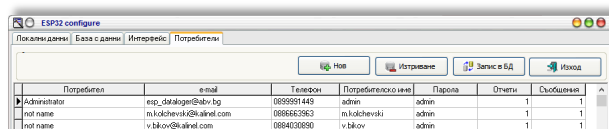
При възникване на грешка контролера отговаря с телеграмата:

#WIP:ERROR

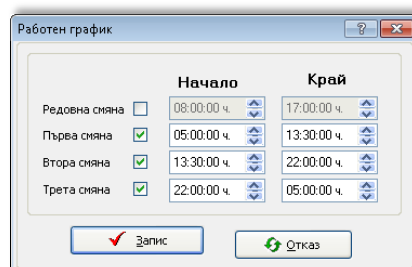
Настройват се следните групи параметри:

- Параметри за връзка по WiFi и Ethernet;
- Параметри за връзка със сървъра;
- Настройка на броя активни канали;
- Настройване на видът на канала броячен и по ниво;

- Настройване на работните периоди;
- Сверяване часовника на контролери (RTC). Ако контролера е свързан с интернет часовника се сверява автоматично от сървъра pool.ntp.org;
- Тестване на комуникацията;
- Тестване на SD картата;
- Рестартиране на контролер;
- Активиране на процес по четене на данни от SD картата.
- Начална инициализация на контролера;
- Задаване на лица и адреси за получаване на служебни съобщения, отчети и известяване за прекъсване и възстановяване на комуникацията с контролерите (фиг. 7) и др.



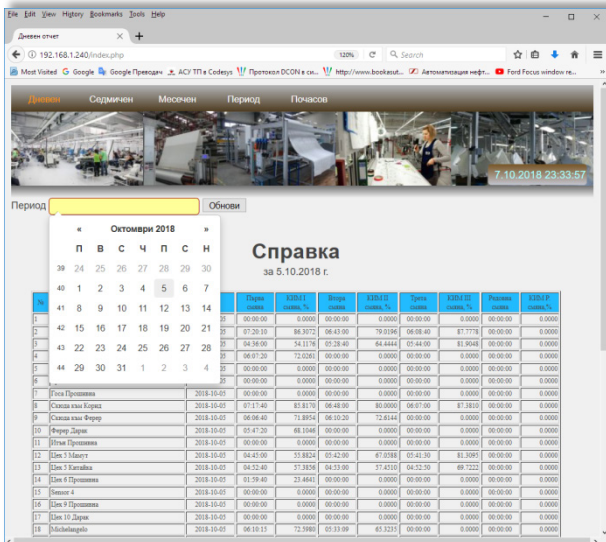
Фиг. 7. Адреси за получаване на съобщения и отчети



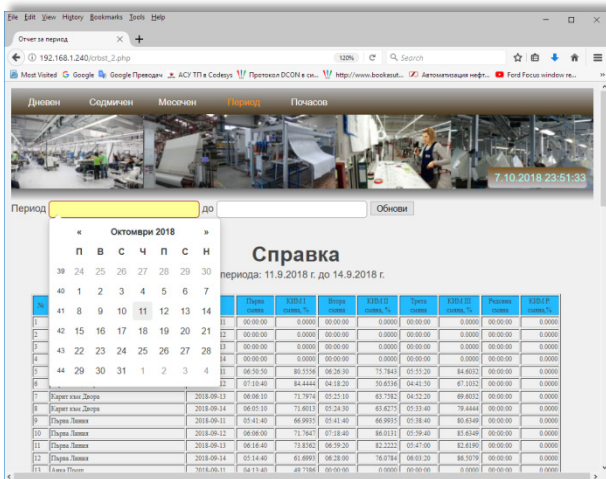
Фиг. 8. Работни графици

Част от изброените параметри се отнасят за конкретната работа на контролерите, друга част например работните смени (фиг. 8) е необходима за сървърното приложение. Информацията за настройките може да се съхранява локално на компютъра от който се извършва настройката или в базата с данни. Позволено е синхронизиране на локалните и сървърните настройки.

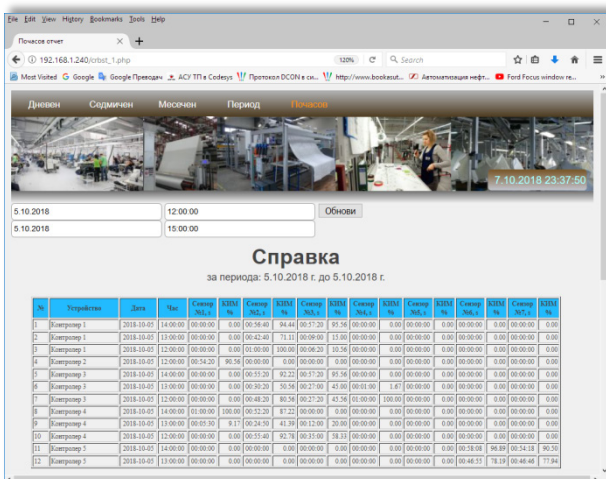
За улесняване работата на ръководителите и служителите на фирмата, е изградено WEB приложение позволяващо генериране освен на дневен, седмичен и месечен отчет (фиг. 9) и на справка за период (фиг.10), и почасова справка (фиг. 11). Тези две допълнителни справки могат да се използват при оптимизиране на работните графици и разпределението на работниците в смените.



Фиг. 9. Дневен отчет



Фиг. 10. Справка за период



Фиг. 11. Почасова справка

ЗАКЛЮЧЕНИЕ

Целта на направеното изследване се заключава в решаването на проблеми по разработване на система за автоматизирано отчитане коефициента на използване на производствените мощности в реално време. Предложеното техническо решение за дистанционен контрол и мониторинг на производствения процес е апробирано и внедрено в текстилно предприятие. Получената информация се използва за контрол на производствения процес в реално време и определяне на трудовите възнаграждения на работниците.

REFERENCE

- [1] Shaul, L., D. Tauber, Critical Success Factors in Enterprise Resource Planning Systems: Review of the Last Decade. ACM Computing Surveys 45(4), 2013, 35 pages.
- [2] Dochev D., M.Velev, I. Iliev, Company Economy, Marinel,Sofia, 2002, in bulgarian
- [3] Internet_Things [Electronic Resource]. Available at: https://en.wikipedia.org/wiki/Web_web.
- [4] Platform with iOs and Android Apps to Control Arduino, ESP8266, Raspberry Pi and Similar Microcontroller Boards over the Internet [Electronic Resource]. Available at: <https://github.com/blynkkk/blynk-server>.
- [5] Internet Of Your Things system Blynk [Electronic Resource]. Available at: <https://www.blynk.cc/>
- [6] Yordanov S. G.Mihalev, Application of esp32 in automated real-time production capacity utilization systems, International Conference Automatics and Informatics'18, 04-06.10.2018. Sofia, Bulgaria, ISSN 1313-1869,1313-1850, pp. 157-161

БЛАГОДАРНОСТ

1. Докладът се публикува във връзка с проект № 1710E/2017
2. Резултатите публикувани в доклада са свързани с НИР по проект № 1710E/2017 към ФЕЕ