## **ТЕХНОЛОГИЧНО УЧИЛИЩЕ ЕЛЕКТРОННИ СИСТЕМИ**

**към ТЕХНИЧЕСКИ УНИВЕРСИТЕТ - СОФИЯ**

**ДИПЛОМНА РАБОТА**

Тема: Система за измерване на метеорологични показатели

Дипломант: Научен ръководител:

*Камен Стефанов* доц. д-р инж. Ст. Стефанова

**СОФИЯ**

**2022**

****

Дипломантът се е справил много добре. Успешно е изпълнил е поставените задачи за проучване на различни видове метеорологични сензори, протоколи за комуникация, платформи за разработка. Избор на база данни за съхранение на снетите данни. Проектиране на печатна платка. Разработка на алгоритъм за управляващ софтуер.

Предлагам за рецензент:

маг. инж. Цветелин Маринов

email: tsvetelin.d.marinov@gmail.com

0887106657

Дата: 31.03.2022 Подпис:

УВОД

Откакто системите за измерване на метеорологични показатели са създадени, те допринасят много за предоставянето на точно измерени метеорологични данни, чрез които може:

* да се правят метеорологични прогнози, нещо от което всеки човек и бита му зависи
* да се предвиждат природни стихии, които могат да навредят на хората
* да се наблюдават и анализират аномалии в природата

Такъв тип информация е от особена важност за хора, чиито начин на живот е много зависим от състоянието на времето, например ако те са земеделци. Тогава за тях всеки ден е от значение, затова източник, от който може да се извлече сигурна и точна информация, би бил изключително важен за производството им.

Поради това в тази дипломна работа ще се разработи автономна система за измерване на метеорологични данни, която ще може да предоставя на потребителя информация отговаряща на региона в който се намира системата, както и направен анализ на снетата информация от сензорите, за да може потребителя да види конкретните данни, които го интересуват- влажност на въздуха, налягане, температура.

**Използвани термини и съкращения**

CSS- Chirp Spread Spectrum

IEEE- Institute of Electrical and Electronics Engineers

IoT- Интернет на нещата

ZigBee- безжична технология за комуникация

LoRaWAN- Long Range Wide Area Network

LNS- LoRa Network Server

GPS- Global Positioning System

IP- Internet Protocol

CRC- Cyclic Redundancy Check

RSSI- Received Signal Strength Indicator

AES- Advanced Encryption Standard

OTAA- Over The Air Activation

DevEUI- уникален сериен идентификатор на крайно устройство

AppKey- приложен ключ за криптиране

NwkKey- мрежови ключ за шифроване)

Bluetooth LE- Bluetooth Low-Energy

DevKit- Development kit

SQL- Structured Query Language

Beaconing- начин на комуникация при клас Б LoRa устройства

Глава 1

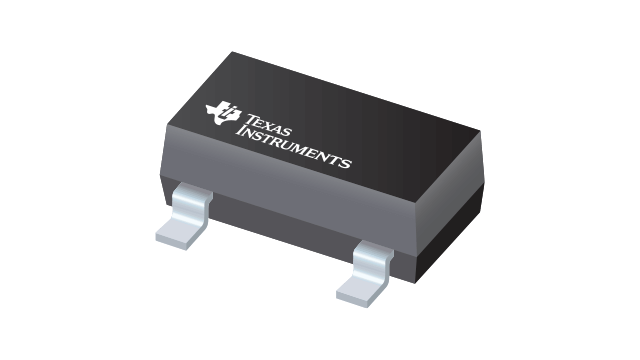
***Преглед на съществуващи системи и управляващи платформи***

### **1.1 . Видове сензори**

Служейки си с набор от сензори, една станция може да измерва различни параметри- температура и влажност на въздух, налягане, скорост и посока на вятър и т.н. В зависимост от начина по който се измерват тези показатели, сензорите могат да бъдат аналогови и цифрови.

#### **1.1.1 Аналогови сензори**

Тези сензори извеждат постоянен аналогов сигнал или ниво на напрежение, пропорционално на измерваната величина. Пример за такъв аналогов измервателен уреди може да бъде аналогов темп. сензор- TMP235 или пък живачния термометър и циферблатния термометър, тоест електрически неутрални измервателни уреди.

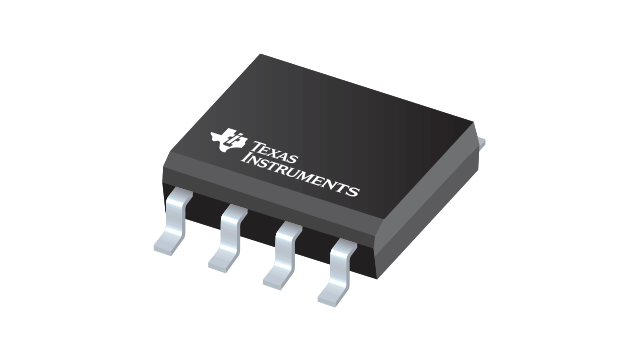
****

*Фиг. 1.1. Texas Instruments TMP235, аналогов темп. сензор*

#### **1.1.2. Цифрови сензори**

Изходът им е под формата на нули и единици.

Цифровите сензори са вид сензори, при които измерваната информация се преобразува в цифров сигнал.



*Фиг. 1.2. Texas Instruments TMP117, цифров темп. сензор*

**Таблица 1.1. Сравнение на аналогов и цифров температурен сензор [1],[2]**

|  | **Параметри** | **TMP117 Цифров температурен сензор** | **TMP235 Аналогов температурен сензор** |
| --- | --- | --- | --- |
| 1. | **Точност на измерване** | от ±0.05°C до ±0.1°C | от ±0.5°C до ±1°C |
| 2. | **Интерфейс** | SMBus, I2C | - |
| 3. | **Консумация** | 3.5µA | 9µAn |
| 4. | **Работен температурен диапазон** | от -55 °C до 150 °C | от -40 °C до 125 °C |
| 5. | **Режим на единично снемане**  **(One-shot-mode)** | Има, трае 15.5ms | - |
| 7. | **Размери** | 2.00 mm x 2.00 mm (WSON Package) | 2.00 mm x 1.25 mm |
| 1.53 mm x 1.00 mm (DSBGA Package) |
| 8. | **Приложение** | * Електронни термометри * Метеорологични станции * Термостат * Автомобилно тестово оборудване * Хладилната индустрия * Топломери и газомери * Температурен трансмитер * Медицински животоподдържащи системи | * Мрежова инфраструктура * Автомобилни инфоразвлекателни системи * Заводска автоматизация * Температурен трансмитер |
| 9. | **Цена** | 24,00 лв | 4,30 лв |

В таблицата по-горе са сравнени най-основните параметри на двата сензора. От сравнението се вижда, че цифровият сензор (TMP 117) превъзхожда аналоговия (TMP 235) по всички параметри освен цената. Но по-високата цена е в полза на по-голямо качество. Ако все пак цената е твърде голяма, има и сравнително по-евтини цифрови сензори, но за сметка на това с ниската цена идват и по-голяма неточност (от ±0.25°C до ±0.5°C), по-големи размери (4.90 mm × 3.91 mm), по-ниска издръжливост и т.н.

### **1.2. Протокол за безжична комуникация**

#### **1.2.1 Wi-Fi [13]**

Wi-Fi е група от безжични мрежови протоколи, базирани на стандартите IEEE 802.11, която обикновено се използва за изграждане на локална мрежа и достъп до Интернет, цифровите устройства в мрежата могат да обменят данни чрез радиовълни. Wi-Fi работи на 2,4 GHz и 5 GHz. Това е един от най-широко използваните мрежови протоколи в света, използван в домашни условия, малки и големи офиси, предприятия, съответно за свързване на устройства- настолни компютри, лаптопи, смартфони и др. Wi-Fi също може да се използва за да се осигури обществен достъп до Интернет в кафенета, летища и др.

#### **1.2.2 Zigbee [14]**

Zigbee се използва главно за създаване на мрежи за управление на автоматизация и сензори по стандартите IEEE 802.15.4 за WPAN(low-rate wireless personal area networks). Тази технология може да работи на 868 MHz, 902-928 MHz и 2,4 GHz честоти в зависимост от изискването на приложението. ZigBee поддържа скорост на данни от 250 Kbps, което е подходящо за двупосочна комуникация между няколко микроконтролера. Технологията намира много голямо приложение в сферата на Интернет на нещата (IoT).

#### **1.2.3 Bluetooth [12]**

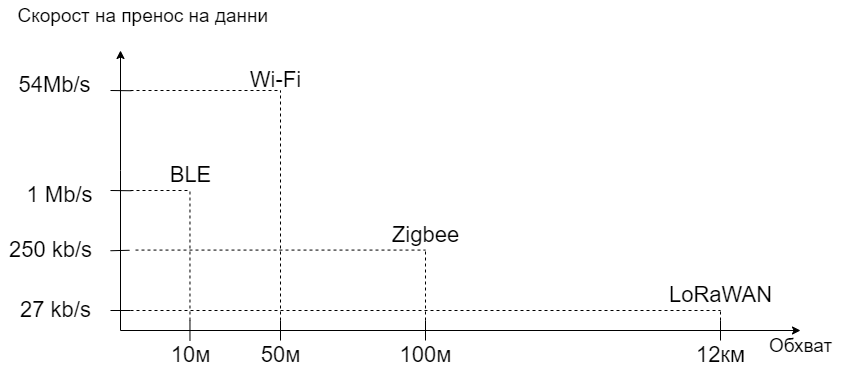
Bluetooth осигурява безжичен начин за свързване и пренос на информация на къси разстояния между устройства- мобилни телефони, компютри и др. Устройството, което се разработва в дипломната работа, ще се захранва от батерия, затова всички компоненти се избират така, че да имат възможно по-ниска консумация на ток. Заради това ще бъде разгледан Bluetooth Low Energy, което е версия на Bluetooth с по-ниска консумация и съответно по-подходящо за проекта. BLE (Bluetooth Low Energy) е технология за безжична лична мрежа (PAN), проектирана и предлагана от Bluetooth Special Interest Group, насочена към нови приложения в здравеопазването, фитнеса, индустрии за сигурност и т.н. В сравнение с “класическия” Bluetooth, Bluetooth Low Energy има за цел да осигури значително намалена консумация на енергия и разходи, като същевременно поддържа подобен обхват на комуникация. BLE работи на честота 2,4 GHz.

#### **1.2.3 LoRaWAN [15]**

LoRaWAN е мрежови протокол с ниска консумация, широка зона на обхват, предназначен за безжично свързване на захранвани от батерии устройства към интернет в малки или големи мрежи и е насочен към ключови изисквания на “Интернет на нещата” (IoT), като двупосочна комуникация, сигурност в комуникацията, мобилност и локализация.

#### **1.2.4 Сравнение на протоколи за безжична комуникация**

На фиг.1.4 са съпоставени скоростта на предаване на данни и обхвата на Wi-Fi, Zigbee, Bluetooth, LoRaWAN.



*Фиг.1.3 Сравнение на протоколи за безжична комуникация*

От сравнението се вижда, че Wi-Fi превъзхожда всички останали протоколи по скоростта на пренос на данни, а LoRaWAN по обхват. За целта на упражнението е нужен протокол с голям обхват, затова ще се използва LoRaWAN. 27 kb/s са напълно достатъчни като скорост за конкретната употреба.

#### **1.2.5 LoRa и LoRaWAN[16]**

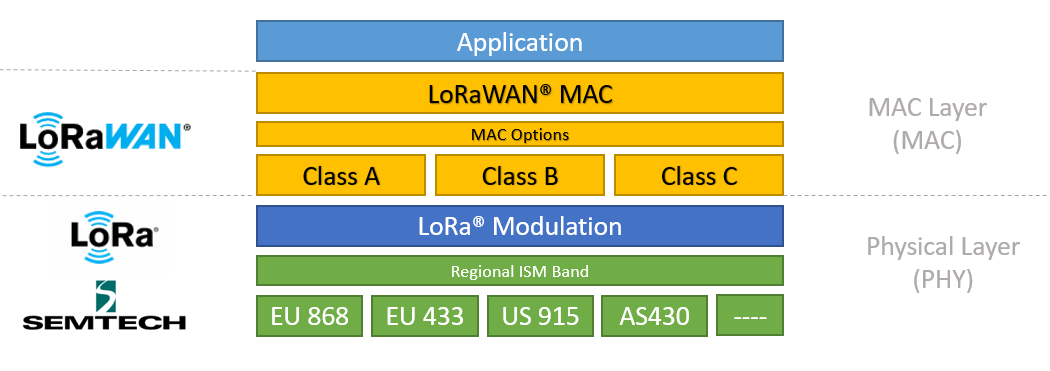
LoRa и LoRaWAN не са термини за едно и също нещо. Както е показано на фиг.1.4, LoRa е физическият слой, т.е. безжичната модулация, използвана за създаване на комуникационна връзка на дълги разстояния, докато LoRaWAN е отворен мрежови протокол, който предоставя сигурна двупосочна комуникация, мобилност и услуги за локализация, стандартизирани и поддържани от LoRa Alliance.

##### **1.2.5.1 LoRa[16]**

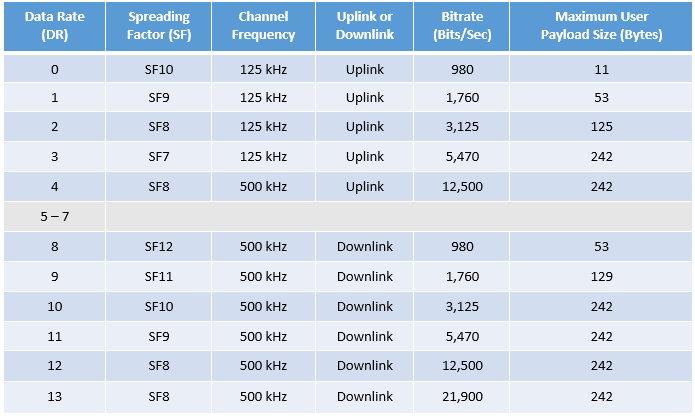
Технология създадена от Semtech за стандартизиране на LPWANs, LoRa осигурява комуникации на далечни разстояния: до пет километра в градските райони и до 15 километра или повече в извънградски райони. Ключова характеристика на базираните на LoRa решения са ниските изисквания за мощност, което позволява създаването на устройства, работещи с батерии, които могат да издържат до 10 години.

* **Радиочестотна модулация**

Semtec патентова техника за модулация с разширен спектър, производна на Chirp Spread Spectrum (CSS). LoRa предлага компромис между чувствителност на приемане от устройствата и скорост на предаване на данни. LoRa работи в канал с фиксирана честотна лента от или 125 KHz, или 500 KHz (за входяща връзка) и 500 KHz (за изходяща връзка). Това позволява на мрежата да увеличи живота на батерията на свързаните крайни устройства, правейки оптимизации в реално време на нивата на консумация и скоростта на предаване на данни. Например, крайно устройство, разположено близо до шлюз, ще предава данни с нисък коефициент на разпространение, тъй като разстоянието от изпращача до приемника е малко, следователно консумацията на устройството ще е редуцирана. Но крайно устройство, което е разположено на поне няколко километра от шлюза, ще трябва да предава с много по-висок коефициент на разпространение. Този по-висок коефициент на разпространение ще увеличи отношението на мощността и интензивността и чувствителността на приемане, съответно и консумацията на устройството, за сметка на по-малка скорост на предаване.



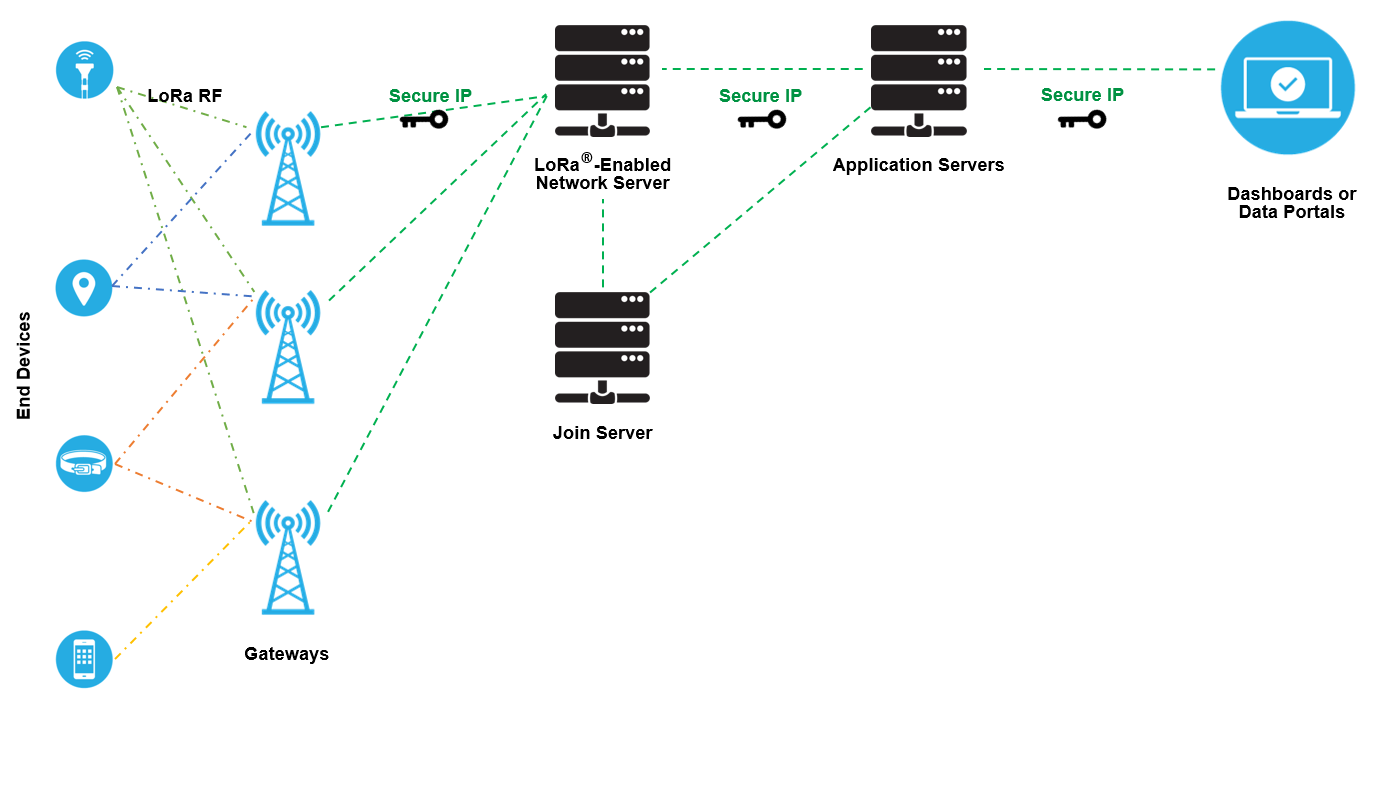
*Фиг.1.4 LoRaWAN технологичен стек*



*Фиг.1.5 LoRa характеристики на модулация*

##### **1.2.5.2 LoRaWAN**

* **Елементи на LoRaWAN мрежата**

Една стандартна LoRaWAN мрежа се състои от: LoRa базирани крайни устройства, LoRaWAN шлюзове, мрежови сървър, приложен сървър, сървър за присъединяване. 

*Фиг.1.6 Стандартна LoRaWAN мрежа и нейните компоненти*

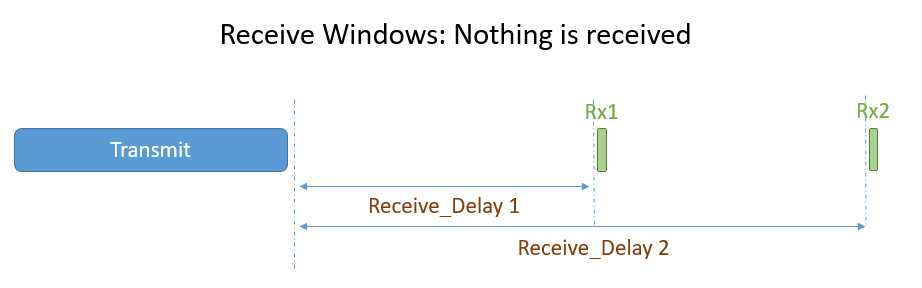
* + **LoRa базирани крайни устройства**

Крайно устройство с активиран LoRaWAN е сензор или актуатор, което е безжично свързано към LoRaWAN мрежа чрез шлюзове с радиовръзка, използвайки LoRa радиочестотна модулация. В повечето приложения крайното устройство е автономно, често захранван от батерии сензор, който дигитализира физическите условия и събитията в околната среда. Типичните случаи на употреба на актуатори включват: улично осветление, безжични брави, предотвратяване на течове и др. Когато се произвеждат, базираните на LoRa устройства получават няколко уникални идентификатора. Тези идентификатори се използват за сигурно активиране и управление на устройствата, за гарантиране на безопасното транспортиране на пакети през частна или публична мрежа и за доставяне на криптирани данни в облака.

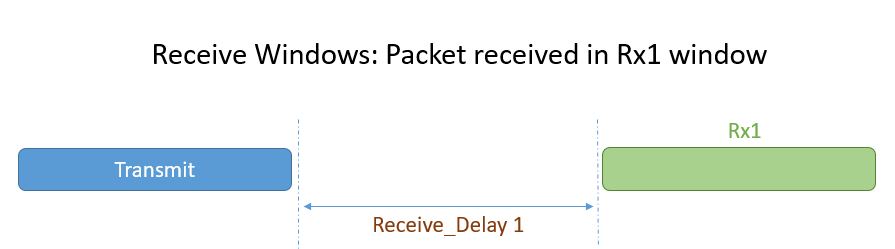
Крайните устройства, базирани на LoRa, могат да работят в три режима, в зависимост от техния клас. Всички такива устройства трябва да поддържат работа от клас А. Устройствата от клас B трябва да поддържат клас A и клас В, а устройствата от клас C трябва да поддържат и трите режима на работа. Тези режими са свързани с това как устройствата комуникират с мрежата.

* + - **Клас А**

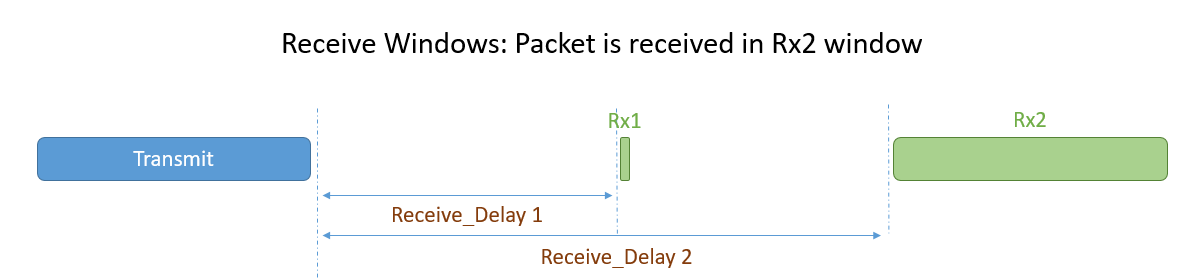
На фиг.1.7 е показан начинът на работа на клас А крайно устройство. В този случай крайното устройство прекарва по-голямата част от времето в неактивно състояние (тоест в режим на заспиване). Когато има промяна в средата, свързана с това, което устройството е програмирано да наблюдава, то се събужда и инициира входяща връзка, предавайки данните за промененото състояние на средата обратно към мрежата (Tx). След това устройството слуша за отговор от мрежата, обикновено за една секунда (въпреки че тази продължителност може да се конфигурира). Ако не получи изходящ пакет по време на този прозорец за получаване (Rx1), той се връща за кратко в режим на заспиване, събужда се момент по-късно, отново слушайки за отговор (Rx2). Ако отново не получи отговор по време на този втори Rx прозорец, устройството се връща в режим на заспиване до следващия път, когато има данни за докладване. Закъснението между Rx1 и Rx2 се конфигурира в зависимост от забавянето от края на входящата сесия. На фигури 1.Х, 1.Х, 1.Х са показани тези комуникационни модели.



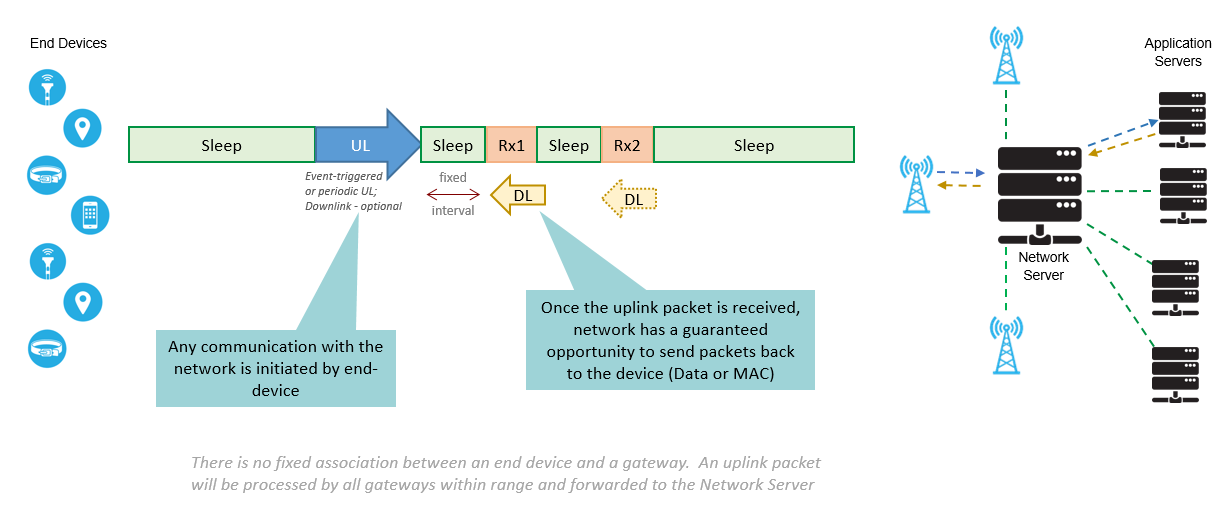
*фиг.1.7 Устройството не получава отговор от мрежата*



*фиг.1.8 Получаване на отговор от мрежата във времеви прозорец RX1*



*фиг.1.9 Получаване на отговор от мрежата във времеви прозорец RX2*

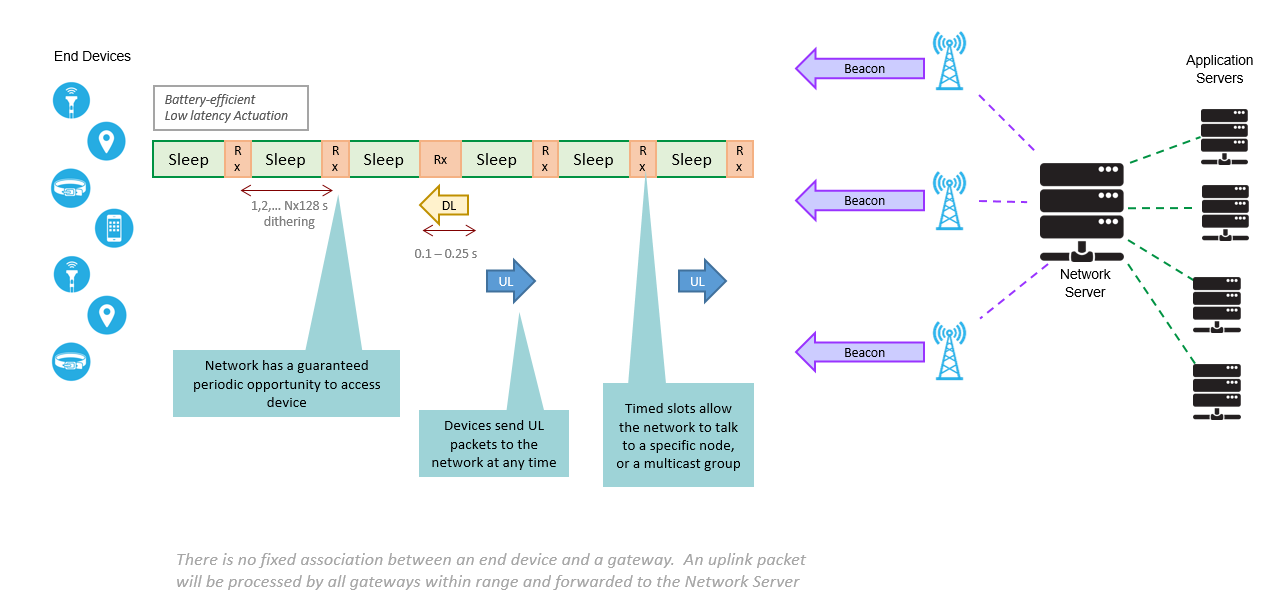


*фиг.1.10 Принцип на работа на Клас А крайно устройство*

* + - **Клас В**

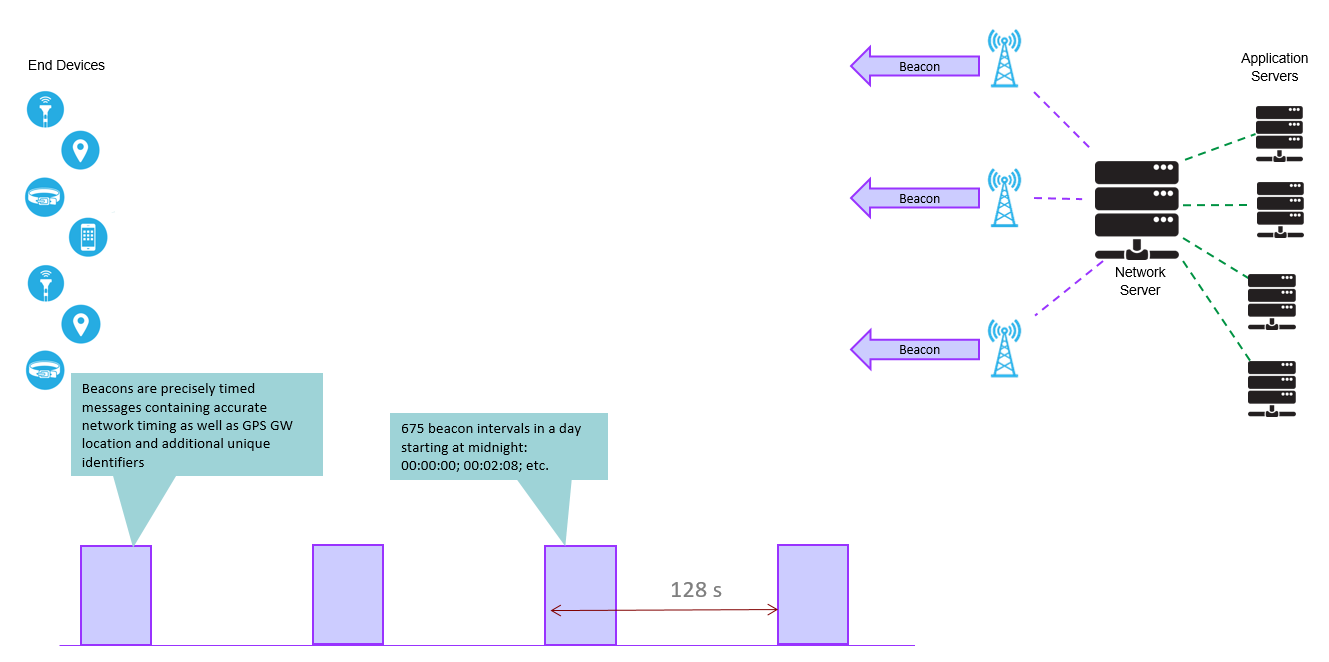
Подобрение на Клас A, LoRaWAN Клас B предлага планирани, фиксирани възможности за крайно устройство да получава изходящи връзки от мрежата, което прави крайните устройства от клас B подходящи както за сензори за наблюдение, така и за актуатори. Всички базирани на LoRa крайни устройства стартират в режим клас А, обаче, устройствата, програмирани със стек от клас B по време на производството, могат да бъдат превключени в клас B режим от приложния слой. Крайните устройства в Class B режим осигуряват планирани прозорци за получаване, в допълнение към тези, които се отварят всеки път, когато към сървъра се изпраща входяща информация от клас A.

За да работи режимът на комуникация от клас B, е необходим процес, наречен beaconing. По време на beaconing процеса, синхронизиран във времето beacon трябва да се излъчва периодично от мрежата през шлюзовете, както е показано на фиг.1.11. Крайното устройство трябва периодично да получава един от тези мрежови beacons, така че да може да изравни своята синхронизираща референция с мрежата .



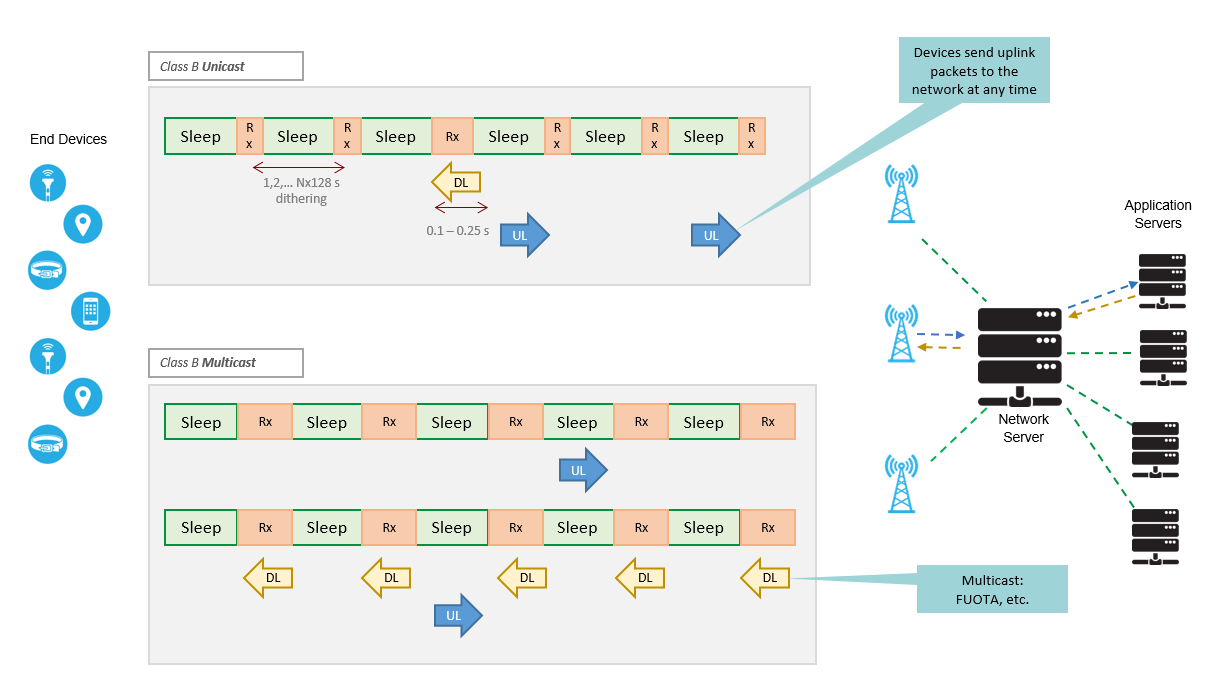
*фиг.1.11 Клас В beaconing*

Устройствата използват beacon, за да сверяват вътрешните си часовници с мрежата. Устройствата не трябва да обработват всеки beacon, ако устройството вече е сверено. В повечето случаи е достатъчно сверяване няколко пъти на ден, като това ще има и минимално въздействие върху живота на батерията, както е показано на фиг.1.12.

****

*фиг.1.12 Периодичен Клас В beaconing за синхронизация на устройства*

Въз основа на тайминга на beacon, крайните устройства могат периодично да отварят времеви прозорци за приемане (пинг слотове). Всеки от тези пинг слотове може да се използва от мрежовата инфраструктура за иницииране на изходяща комуникация, както е показано на фиг.1.12. За да може една LoRaWAN мрежа да поддържа устройства от клас B, всички LoRaWAN шлюзове в тази мрежа трябва да имат вградена GPS синхронизация, така че всички те могат да бъдат синхронизирани.

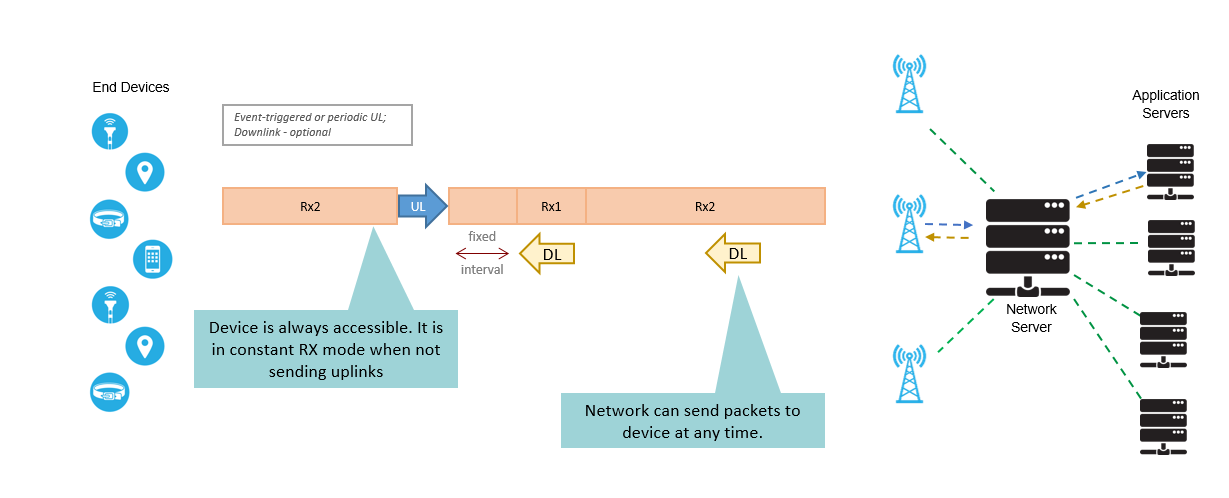
****

*фиг.1.13 Клас В пинг слотове*

* + - **Клас С**

На фиг.1.14 е показан начинът на работа на клас С крайно устройство. Клас C винаги е „включен“, тоест не се захранва от батерия. Устройствата от клас C включват такива неща като- улично осветление, електрически измервателни уреди и т.н. Тези устройства винаги слушат за изходящи съобщения, освен ако не предават входящи. В резултат на това те предлагат най-ниската латентност за комуникация от сървъра до крайно устройство.

Крайните устройства от клас C използват същите два прозореца за получаване като устройствата от клас A, но те не затварят прозореца Rx2, докато не изпратят следващото предаване обратно към сървъра. Следователно те могат да получат изходяща връзка в прозорец Rx2 почти по всяко време. Кратък прозорец със същата честотата и скорост на данни като Rx2 също се отваря между края на предаването и началото на прозореца за получаване Rx1.



*Фиг. 1.14 Принцип на работа на Клас С крайно устройство*

* + **LoRaWAN шлюзове**

LoRaWAN шлюзът получава модулирани от LoRa съобщения от всяко крайно устройство, което е на достатъчно близко разстояние, така че съобщенията да се изпращат и получават успешно, и препраща тези съобщения с данни към LoRaWAN мрежовия сървър (LNS), който е свързан чрез IP опорна мрежа. Няма фиксирана връзка между крайно устройство и конкретен шлюз. Вместо това един и същ сензор може да се обслужва от множество шлюзове около него. С LoRaWAN всеки входящ пакет, изпратен от крайното устройство, ще бъде получен от всички шлюзове в обсега. Това значително намалява процента на грешки при пакетите (тъй като шансът, поне един шлюз да получи пакета, е много голям ) и позволява геолокация с ниска цена (ако приемем, че въпросните шлюзове са с възможност за геолокация). IP трафикът от шлюз към мрежовия сървър може да бъде прехвърлен чрез Wi-Fi, Ethernet кабелна връзка или чрез клетъчна връзка. LoRaWAN шлюзовете работят изцяло на физическия слой и по същество не са нищо друго освен препращачи на LoRa радио съобщения. Те проверяват само целостта на данните на всяко входящо LoRa съобщение. Ако целостта е нарушена, тоест, ако CRC (Cyclic Redundancy Check) е неправилен, пакетът ще бъде отхвърлен. Ако е правилен, шлюзът ще го препрати към мрежовия сървър, заедно с някои метаданни, които включват нивото на RSSI (Received Signal Strength Indicator) на съобщението, както и времеви печат, който не е задължителен. За LoRaWAN изходяща връзка, шлюзът изпълнява заявки за предаване, идващи от мрежовия сървър, без прекъсване от полезни данни. Тъй като множество шлюзове могат да получат едно и също LoRa радио съобщение от едно крайно устройство, мрежовият сървър извършва дедуплициране на данни и изтрива всички копия. Въз основа на RSSI нивата на идентичните съобщения, мрежовият сървър обикновено избира шлюза, който е получил съобщението с най-добрия RSSI при изпращане на съобщение за изходяща връзка, тъй като този шлюз е най-близо до въпросното крайно устройство.

* + **Мрежови сървър**

Мрежовият сървър управлява цялата мрежа, динамично контролира мрежовите параметри, за да адаптира системата към постоянно променящите се условия и установява сигурна 128-битов AES свързаност за транспортиране както на данни от край до край (от крайно устройство до апликациите на крайните потребители в облака), така и за контрол на трафика, който преминава от крайното устройство до мрежовия сървър и обратно. Мрежовият сървър гарантира автентичността на всеки сензор в мрежата и целостта на всяко съобщение. В същото време мрежовият сървър не може да види или да получи достъп до апликационните данни.

Като цяло всички LoRaWAN мрежови сървъри споделят следните характеристики:

* Проверка на адреса на устройството
* Потвърждение за получени съобщения
* Адаптиране на скоростите на данни с помощта на ADR протокол
* Отговаряне на всички заявки от MAC слоя, идващи от устройството
* Препращане на входящ приложен полезни данни към съответните приложни сървъри
* Подреждане на изходящи полезни данни, идващи от всеки приложен сървър и отиващи към всяко устройство, свързано към мрежата
* Препращане на съобщения за заявка за присъединяване и за приемане на заявка за присъединяване между устройствата и сървъра за присъединяване
  + **Приложен сървър**

Приложните сървъри са отговорни за сигурната обработка и управление на данните от данните снети от сензорите. Те също така генерират всички изходящи полезни данни от приложния слой, свързани към крайните устройства.

* + **Сървър за присъединяване**

Сървърът за присъединяване управлява процеса на активиране по въздух (OTAA) на крайни устройства, чрез който те се присъединяват към мрежата. Сървърът за присъединяване съдържа информацията, необходима за обработка на входящи рамки със заявка за присъединяване и генериране на изходящи рамки за приемане на заявка за присъединяване. Той сигнализира на мрежовия сървър кой приложен сървър трябва да бъде свързан към крайно устройство и да обменя ключовете за криптиране и декриптиране. Той комуникира ключа за мрежовата сесия на устройството с мрежовия сървър и ключа за приложната сесия към съответния приложен сървър. За тази цел сървърът за присъединяване трябва да съдържа следната информация за всяко крайно устройство под негов контрол:

-DevEUI (уникален сериен идентификатор на крайно устройство)

-AppKey (приложен ключ за криптиране)

-NwkKey (мрежови ключ за шифроване)

-Идентификатор на приложния сървър

-Профил на услугата за крайно устройство

### **1.3. Платформи, подходящи за реализиране на системата**

В днешно време готовите платформи не са новост. Тези малки и удобни за употреба устройства са намерили различно приложение в ежедневието на много хора. Спомагат за обучението на ученици за програмиране на вградени микрокомпютърни системи, за по-лесно отстраняване на грешки в разработката на различни видове устройства и др.

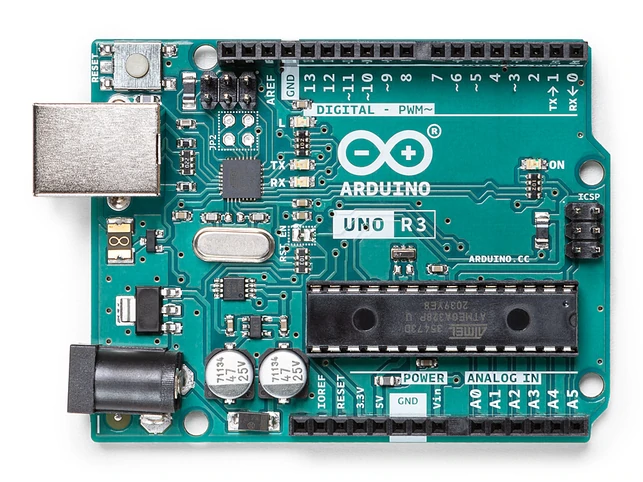
#### **1.3.1. Arduino [6]**

Ардуино е малка платка, изключително подходяща за въведение в програмиране на вградени микрокомпютърни системи. Снабдена с мощни ATmega микроконтролери, Ардуино може да управлява всички процеси на роботи, IoT системи и интерактивни проекти. С помощта на различни сензори (напр. за звук, светлина, докосване, движение и др.) Arduino може да отчита промени в заобикалящата среда и да задейства някакви актуатори- серво мотор, хидравлика, водна помпа и т.н. Освен със сензори и задвижващи механизми Ардуино може да комуникира двупосочно и със софтуерни програми като Flash, Pure Data (PD), Processing и др.

Най- популярната платка от многото серии на компанията е Arduino Uno. Тя е снабдена с:

* микроконтроллер- ATmega328P
* 14 цифрови извода (могат да приемат информация и да изпращат информация)
* 6 аналогови входни извода (могат да приемат показания от аналогови сензори)
* 6 аналогови изходни извода (използват ШИМ за да маскират цифровите изходи като аналогови.)

На фиг.1.3 е показана Ардуино платформата.



*Фиг.1.15 Arduino платформа*

#### **1.3.2. ESP32 [10]**

ESP32 платформата, продукт на ESPRESSIF, чрез интегрирането на Wi-Fi, Bluetooth и Bluetooth LE се гарантира широк спектър от приложения. Платформата има вградени: балун изпълняващи ролята на съгласуващ импедансите елемент между антената и радиоапаратурата при честоти в широк диапазон, усилвател на мощност, усилвател за приемане с нисък шум, филтри и модули за управление на мощността. Всички модули от серията ESP32 имат широк работен температурен диапазон от -40°C до 105°C и са подходящи за разработване на IoT проекти със своя 4-слоен дизайн, който покрива редица стандарти- FCC, CE-RED, SRRC, IC, KCC & TELEC.

Едни от по-полярните серии на ESP32 са ESP32-S серрите с модул ESP32-S2-DevKitM-1. Модулите от тази серия разполагат с редица интерфейси - слот за SD карта, USB, UART, SPI, I2C, I2S, LED PWM (интерфейс за управление на LED светлини), Motor PWM (интерфейс за управление на електромотори). ESP32-S2-DevKitM-1 конкретно разполага с Extensa микроконтролер, с 42 извода:

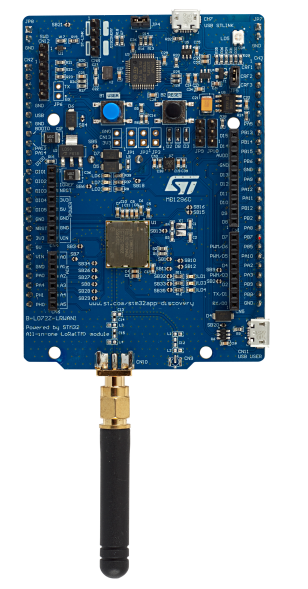
* 37 GPIO извода (могат да приемат информация и да изпращат информация), 21 от които са с АЦП(Аналогово -Цифров-Преобразувател) и 2 с ЦАП (Цифрово -Аналогов-Преобразувател)
* 4 извода за земя и захранване
* 1 рестартиращ извод



*фиг.1.16 ESP32 платформа*

#### **1.3.3. STМ 32 [8]**

Комплектите за разработка (DevKit) на ST освобождават инженерите от нуждата да разработват свои собствени платки, за да изпробват интегралните схеми на ST. Базираният, като такъв на компютър, графичен интерфейси позволява на разработчиците да конфигурират и наблюдават важни електрически параметри. Освен това е наличен пълен набор от документи и ресурси, включително схеми и сметки за материали, както и справочни ръководства, за да помогне на инженерите да разработят своите приложения. Една конкретна платка от STM32 сериите е STM32L0. Тази платка поддържа технологии с ниска консумация на електричество за безжична комуникация на базата на радиовълни, като LoRa и Sigfox и е снабдена с STM32L072CZ микроконтролер и SX1276 приемо-предавател. Този модул разполага с USB 2.0 интерфейс, UART, I2C и SPI. На фиг.1.5 е показана STM32 платформата.



*фиг.1.17 STM32 платформа*

#### **1.3.4. Съпоставка между различните платформи**

В таблицата са представени най-подходящите модули, за реализиране на този проект, подбрани от Arduino, ESP32 и STM32.

**Таблица 1.2. Сравнение на платформи**

|  | **Параметри** | **Arduinoi Nano 33 IoT** | **ESP32-S2-DevKitM-1** | **STM32LO** |
| --- | --- | --- | --- | --- |
| **1.** | **Действие** | SOM (System on a module) | SOM (System on a module) | SOM (System on a module) |
| **2.** | **Безжична комуникация** | Bluetooth, Wi-Fi | Wi-Fi | LoRa, SigFox |
| **3.** | **Скорост на процесор**  **(CPU clock speed)** | 48 MHz | 240 MHz | 32 MHz |
| **4.** | **Флаш памет** | 256 KB | 4 MB | 192 KB |
| **5.** | **GPIO изводи** | Да, 23 | Да, 37 | Да, 40 |
| **6.** | **Интерфейс за**  **захранване/програмиране** | Micro USB | Micro USB | Micro USB |
| **7.** | **Интерфейси: SPI, I2C, UART** | Да | Да | Да |
| **8.** | **Консумация**  **на електричество** | 47 mA (BLE), 110 mA (Wi-Fi) | 310 mA\* | 7,4 mA |
| **9.** | **Поддържани езици за**  **програмиране** | C/C++ | C/C++ | C/C++ |
| **10.** | **Цена** | 31,49 лв | 15,74 лв | 77,98 лв |
| **11.** | **Работен температурен**  **диапазон** | от -40 °C до 80 °C | от -40 °C до 85 °C | от -40 °C до 125 °C |

\*При изпращане на данни (TX) с Wi-Fi стандарт- 802.11b

От представената таблица става ясно, че всеки от модулите превъзхожда останалите в някои параметри, но в други отстъпва. Едно от главните изисквания на този проект е комуникацията да се осъществи от протокол с ниска консумация на ток и голям обхват, за тази цел беше избран LoRaWAN, който се поддържа единствено от STM32LO, затова този модул е избран за разработката на дипломната работа. STM32LO изпълва всички изисквания- ниска консумация, поддържа LoRaWAN, достатъчно мощен микроконтролер, достатъчно флаш памет.

### **1.4. Шлюз [18]**

Тъй като протоколът за комуникация ще бъде LoRaWAN, за шлюз ще се използват други STM устройства- P-NUCLEO-LRWAN2 STM32, което е комплект от два модула- NUCLEO-F746ZG и LRWAN\_GS\_LF1.

NUCLEO-F746ZG:

* микроконтролер- STM32F746ZGT6 Arm® Cortex®-M7, 216 MHz, 1-Mbyte Flash памет и 320-Kbyte SRAM
* 3 потребителски светодиода
* 1 потребителски бутон и 1 бутон за нулиране
* Ethernet съвместим с IEEE-802.3-2002
* 32,768 kHz кристален осцилатор

LRWAN\_GS\_LF1:

* Автоматично се адаптира към коефициента на разпространение от SF12 до SF7 във всички 8 канала
* Висока чувствителност до -140 dBm при 300 bit/s
* 6 dBm изходна мощност
* Поддръжка на протокол LoRaWAN® клас A и клас C
* Поддръжка за пренасочване на пакети Semtech
* Поддържа DNS и NTP

### **1.5. Бази данни [17]**

Базата данни е инструмент за събиране и организиране на информация. Базите данни могат да съхраняват информация за хората, продуктите, поръчките или каквото и да било друго. Много бази данни се стартират като списък в програма за текстообработка или електронна таблица. Тъй като списъкът става по-голям, има поява на много несъответствия и грешки в базата от данни. Данните стават все по-трудни за разбиране и има ограничени начини за търсене или издърпване на под набори от данни за преглед. След като започнат тези проблеми, е добра идея да данните да се прехвърлят в база данни, създадена от система за управление на бази данни (СУБД).

#### **1.5.1. Видове бази данни**

Организацията и подредбата на данните може да следва определен модел. Според модела може да се опишат два типа бази данни – релационни и нерелационни. Най-известният и използван до момента тип бази данни е релационният (SQL бази данни). Уеб приложения и решения с отворен код са MySQL, PostgreSQL и SQLite. Другият тип бази данни, използващ нерелационен модел за данните, се нарича NoSQL (non SQL / Not only SQL). Решения за нерелационни бази данни са MongoDB, Redis и други. NoSQL базите данни се използват при работа с огромен обем от данни и приложенията, работещи в реално време.

* Релационните бази, срещани съхраняват данните по предварително структуриран начин – в таблици, подредени в редове (записи) и колони. Между отделните данни и таблици може да се създават връзки (релации). Таблиците и връзките между тях образуват структура, която може да се представи като схема. На фиг.1.7 е показан пример за заредена релационна база данни. SQL базите се използват масово – при малки обеми информация например за уеб сайт с две страници, до големи уеб или мобилни приложения, блогове, онлайн магазини и други.

Разрастване(скалируемост)- Базата данни се намира на един сървър. За да се разшири, може да се увеличава мощта и ресурсите на този сървър. Възможно е SQL базата данни да се разпростре на множество сървъри, но имплементацията обикновено е сложна и са нужни повече ресурси и време. И тъй като тези бази не предлагат подобна функционалност по естествен начин, допълнителна разработка ще е нужна, за да може различните хардуерни точки да имитират работата на една база, на един сървър. Допълнителни програмни разработки ще са нужни за управление на логиката и дистрибутирането на заявките за данните между отделните точки, както и за извличането и обединяването на данни от различните сървъри.

* NoSQL базите данни са общо наименование на различни технологии за бази данни, създадени за нуждите на модерните приложения и огромното количество информация, с което те работят. С NoSQL базите данни се решават различни SQL ограничения за:
* лесна мащабируемост върху клъстери от сървъри (хоризонтално мащабиране);
* поддръжка на различни типове структури от данни;
* използване при разработка с гъвкави методологии (agile development).

### **1.5. Захранване**

Литиево-йонните батерии са предпочитан избор за захранване на IoT(система), но тъй като STM32LO модулът има държач за батерии за 3 AAA батерии, за целта на упражнението ще се използва той.

#### **1.6.1. Слънчев панел [19]**

### Слънчевият панел има роля на допълнително захранване на системата. Ще се използва марка OSEPP, модел SC10036. На фигура 1.х е показан панелът.



*Фиг. 1.18 Слънчев панел SC1036*

### **1.7. Среда за разработка на управляващ софтуер [3]**

Тъй като платформата върху която ще се разработва дипломната работа е STM32LO , ще се използва STM32CubeIDE, като среда за разработка на управляващ софтуер. STM32Cube поддържа:

* C и C++ езици за програмиране
* хардуерна абстракция, позволявайки по-голяма портативност за различни STM32 устройства
* API-та [Х] на ниско ниво, чрез които се достига по-голяма производителност и по-дълъг живот на системи, зареждани с батерии
* Много библиотеки за различни сензори и актуатори

### **1.8. Типове метеорологични станции**

Ако на човек му е нужно да получава актуална информация за метеорологичните условия, той може да черпи информация от прогнози за времето, но те предоставят показатели взети за по-голям регион- квартал или град, а не дават конкретна информация, отнасяща се до точното му местоположението. Затова човек може да използва метеорологична станция. Станциите, в зависимост от сензорите, използвани за измерване на метеорологични параметри, се делят на два вида- цифрови и аналогови.

#### **1.8.2. VA Bresser метеорологична станция [20]**

* VA Bresser е един пример за метеорологична станция, която разполага с цифров дисплей и цифрови сензори. Тя може да измерва вътрешна, външна температура, влажност и да прави прогноза за времето (прогнозата си обновява на 12 часа). На Фиг. 1.1. е показана съответната станция. Сензорите на тази станция са безжични и имат обхват на предаване 60 м. Тази станция е подходяща за домашна употреба, но не е толкова надеждна.

****

*Фиг. 1.19. Цифрова станция*

Глава 2  
***Проектиране на блокова схема на система за измерване на метеорологични данни***

### **2.1. Функционални изисквания към системата**

Системата, която се проектира, има за цел да предостави информация за околната среда, чрез сензор и други допълнителни компоненти. За целта ще бъдат направени проучвания за сензора, който ще измерва необходимите метеорологични параметри. Също така ще бъде направено проучване за управляваща микроконтролерна платформа. След като бъдат избрани компонентите за метеорологичната станция, ще бъде проектирана принципна електрическа схема на допълнително устройство, което ще изпълнява функцията на допълнително захранване за микроконтролерната платформа и физическа връзка между сензорите и управляващата платформа. Впоследствие ще бъде проектирана платка за допълнителното устройство. Следващото функционално изискване е проектиране на алгоритъм за управляващ софтуер, който ще се изпълнява от микроконтролерната платформа.

### **2.2 Обща блокова схема и описание на отделните блокове**

На фиг.2.1 е показана общата блокова схема на проектираната система.



*Фиг. 2.1 Обща блокова схема на системата*

### **2.2.1. Описание на отделните блокове**

**Блок “Управление”-** захранва се от блок “Захранване” ислужи за управление на цялата система, като приема информация от входния блок “Измерване” и я предава на блок “Приемане и изпращане на снети данни”.

**Блок “Измерване”-** служи за измерване на редица метеорологични показатели (температура, влажност на въздуха; налягане; скорост на вятъра), които изпраща към блок “Управление”.

**Блок “Приемане и изпращане на снети данни”-** приема информацията от блок “Управление” и я препраща към отдалечената база данни.

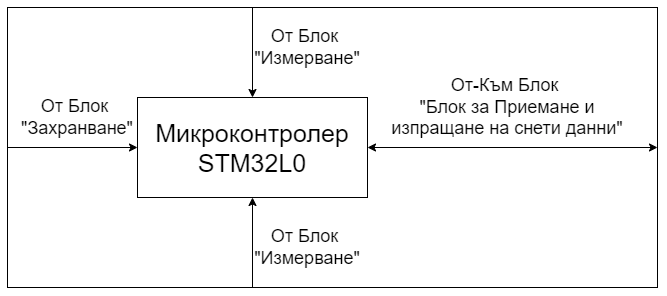
**Блок “Захранване”-** служи за захранване на блокове “Измерване”, “Управление”, “Приемане и изпращане на снети данни”.

**Отдалечена База Данни-** инструмент за събиране и организиране на информация, с възможност за последващ анализ на данните.

**2.3. Блокови схеми на отделните блокове**

### **2.3.1. Блок “Управление”**

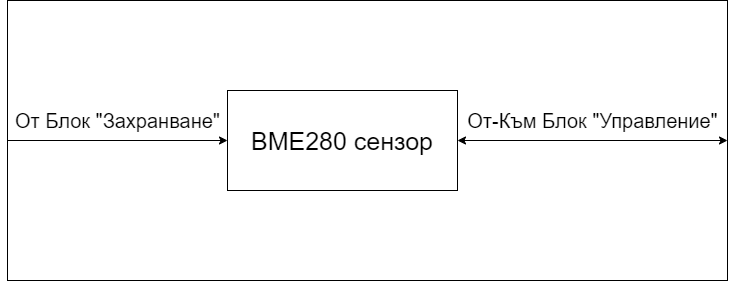
Блок “Управление” се състои от управляващ микроконтролер STM32L0, който ще контролира начина на работа на системата. На Фиг.2.2. е показана вътрешната структура на съответния блок. На нея се вижда, че управляващият микроконтролер получава снетите от сензорите в блок “Измерване” данни, след което ги изпраща към блок “Приемане и изпращане на снети данни”. Микроконтролерът се захранва от блок “захранване”.

****

*Фиг. 2.2 Вътрешна структура на Блок “Управление”*

### **2.3.2. Блок “Измерване”**

Блок “Измерване” се състои от един сензор- BME280. Той измерва температура, налягане, влажност на въздуха. Измерените параметри се изпращат към Блок “Управление”. Блокът се захранва от Блок “Захранване”.

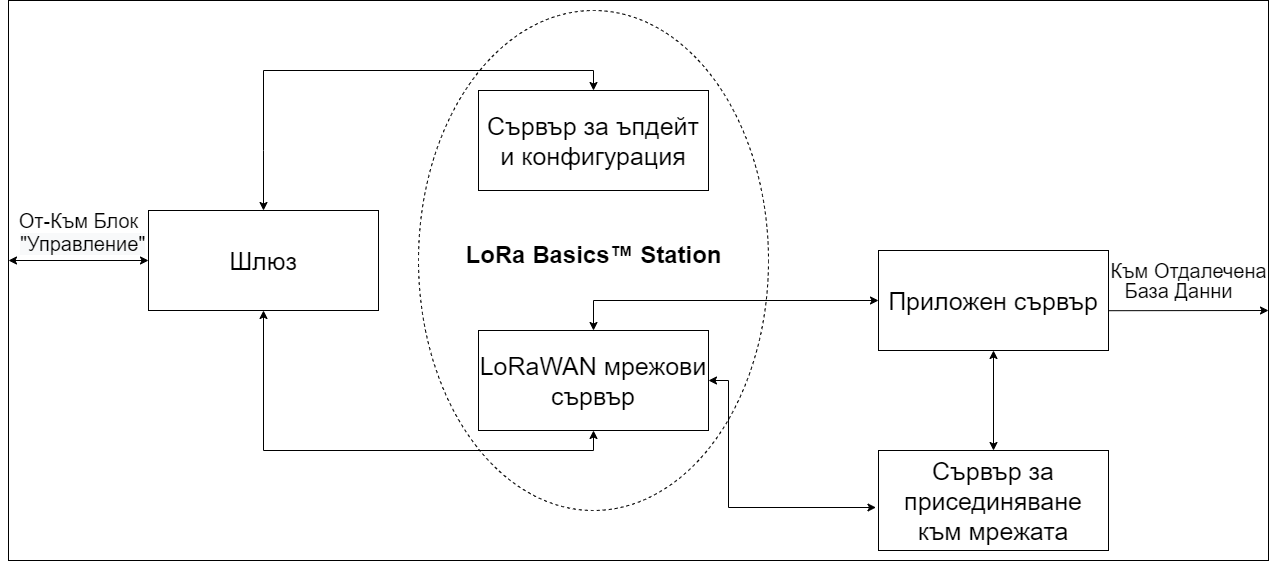


*Фиг. 2.3 Вътрешна структура на Блок “Измерване”*

### **2.3.3. Блок “Приемане и изпращане на снети данни”**

Блок “Приемане и изпращане на снети данни” се състои от: Шлюз,сървър за ъпдейт и конфигурация, LoRaWAN мрежови сървър, Приложен сървър и Сървър за присъединяване към мрежата.

* Шлюз- получава данни от блок “Управление” и ги препраща към сървъра за ъпдейт и конфигурация и LoRa мрежовия сървър
* LoRa Basics™ станция
  + Сървър за ъпдейт и конфигурация- позволява на мрежовия сървър да конфигурира отдалечено шлюзове и да актуализира фърмуера им
  + LoRaWAN мрежови сървър- необходим за изпращане и получаване на данни от LoRaWAN. Установява връзка за предаване на данни между шлюзa и мрежовия сървър.
* Сървър за присъединяване към мрежата- служи за добавяне на крайни устройства към мрежата.
* Приложен сървър- обработва и управлява данните от sensor application data. Генерира цялата изходящи данни от приложния слой.

****

*Фиг. 2.4 Вътрешна структура на Блок “Комуникация”*

### **2.2.4. Блок “Захранване”**

Блок “Захранване” се състои от три AAA акумулаторни батерии и слънчев панел. Батериите са основното захранване на системата.



*Фиг. 2.3 Вътрешна структура на Блок “Захранване”*

Глава 3

***Проектиране на принципна електрическа схема на система за измерване на метеорологични данни***

### **3.1. Избор на CAD система за проектиране**

За проектиране е избрана програмата OrCAD.

### **3.2. Избор на основни компоненти**

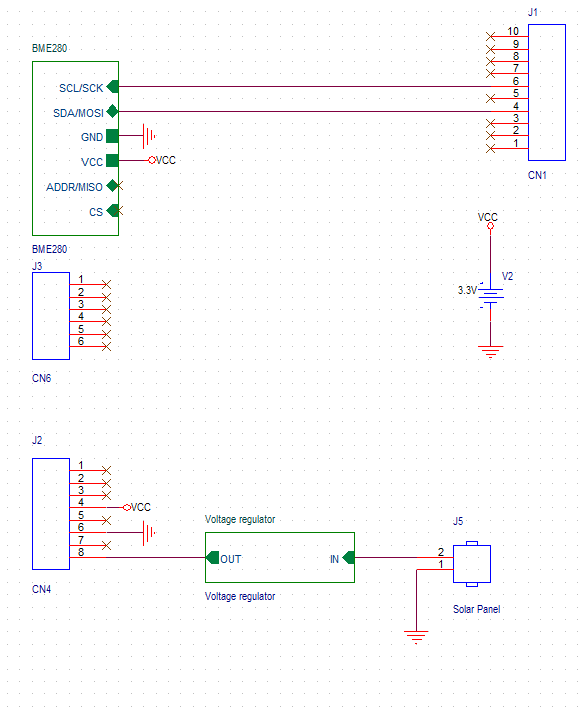
#### **3.2.1 Блок “BME280”**

За измерване на метеорологични показатели е избран сензор, произведен от Bosch, BME280. Размерът му е 26mmx20mm. Входното му напрежение е 3.3V. На фиг. 3.1 е представен сензорът. [21]



фиг. 3.1. BME280

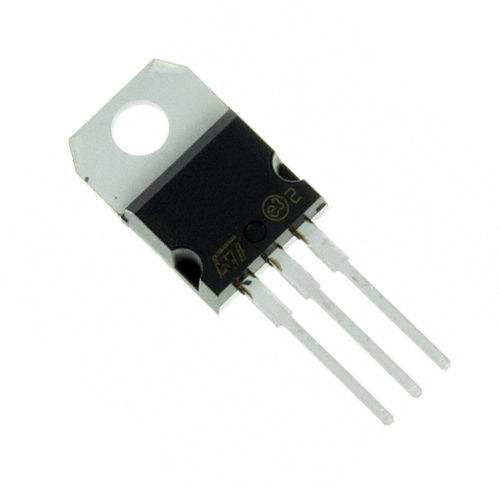
*На фиг. 3.2 е показано схемното означение на сензор BME280.*

**

*фиг. 3.2. Схемно означение BME280*

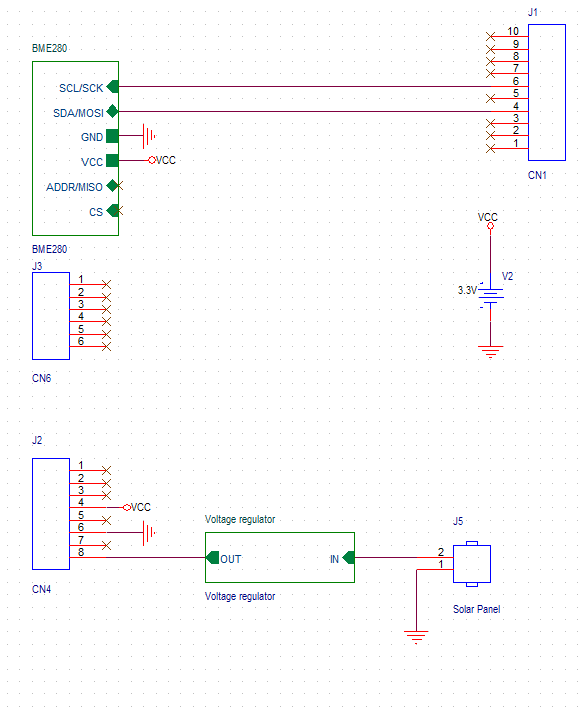
#### **3.2.2 Блок “Voltage regulator” (Регулатор на напрежение)**

За регулиране на напрежението, идващо от захранващите соларни панели, е избран LM7805 произведен от Texas Instruments. Размерът му е 10.67mmx16.5mm. На фиг. 3.3 е показан елементът. [22]



*фиг. 3.3. LM7805*

На фиг. 3.4 е показано схемното означение на LM7805.

**

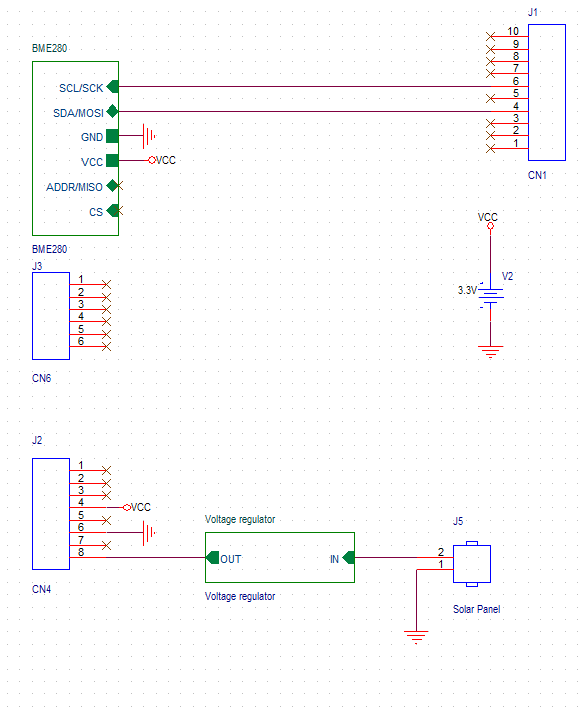
*фиг. 3.4. Схемно означение LM7805*

### **3.3. Проектиране на принципна електрическа схема на системата**

В тази точка ще бъде описана и разгледана принципната електрическа схема на устройството, което ще бъде проектирано за по-добра работоспособност на системата. То ще изпълнява няколко функции: допълнително захранване на микропроцесорната платформа, измерване на метеорологични данни чрез сензори (връзка между микропроцесорната платформа и сензорните).

### **3.3.1 Йерархична структура на устройството**

На фиг. 3.5 е показана йерархичната структура на устройството. Има два йерархични блока: “BME280” и “Voltage regulator”.

**

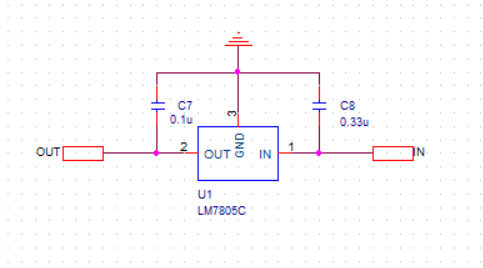
*Фиг. 3.5 Принципна електрическа схема*

### **3.3.2 Йерархичен блок “BME280”**

*A3 схема*

*Фиг. 3.6 Блок “BME280”*

### **3.3.3 Йерархичен блок “Voltage Regualtor”**

**

*Фиг. 3.7 Блок “Voltage Regualtor”*

Глава 4

***Проектиране на печатна платка на система за измерване на метеорологични данни***

### **4.1. Проектиране на печатна платка**

В тази глава ще бъде разгледан начина на проектиране на печатната платка необходима за изграждане на системата.

### **4.2. Избор на корпуси на елементите**

Елементи, чиято каталожна информация не отговаря на съществуващите корпуси, с които разполага системата за чертане на електрически схеми, се чертаят на ръкa. В **таблица 4.1** са представени конкретните елементи и съответстващите им корпуси.

|  | Компонент | Стойност | Корпус |
| --- | --- | --- | --- |
| 1. | R1, R2, R3, R4,  R5, R6, R7, R8 | 4.7K | ERA3 |
| 2. | C1, C4 | 10uF | CNC6 |
| 3. | C2 | 1uF |  |
| 4. | C3 | 22nF | CGA9 |
| 5. | C5, C6, C7 | 100nF | CGA8 |
| 6. | C8 | 0.33uF | CGA6 |
| 7. | Q1B, Q2A, Q3B, Q4B | - | SOG/NDC |
| 8. | U3 | - | SOG/RT |
| 9. | U2 | - | BME |
| 10. | J6 | - | JUMPER100 |
| 11. | J1 | - | BLKCON.100/VH/TM1SQ/W.100/10 |
| 12. | J2 | - | BLKCON.100/VH/TM1SQ/W.100/8 |
| 13. | J3 | - | BLKCON.100/VH/TM1SQ/W.100/6 |
| 14. | J5 | - | BLKCON.100/VH/TM1SQ/W.100/2 |
| 15. | D1 | 10mA | CGA3 |
| 16. | U11 | - | TO220AA/RF1 |

*Таблица 4.1*

### **4.3. Проектиране на печатна платка**

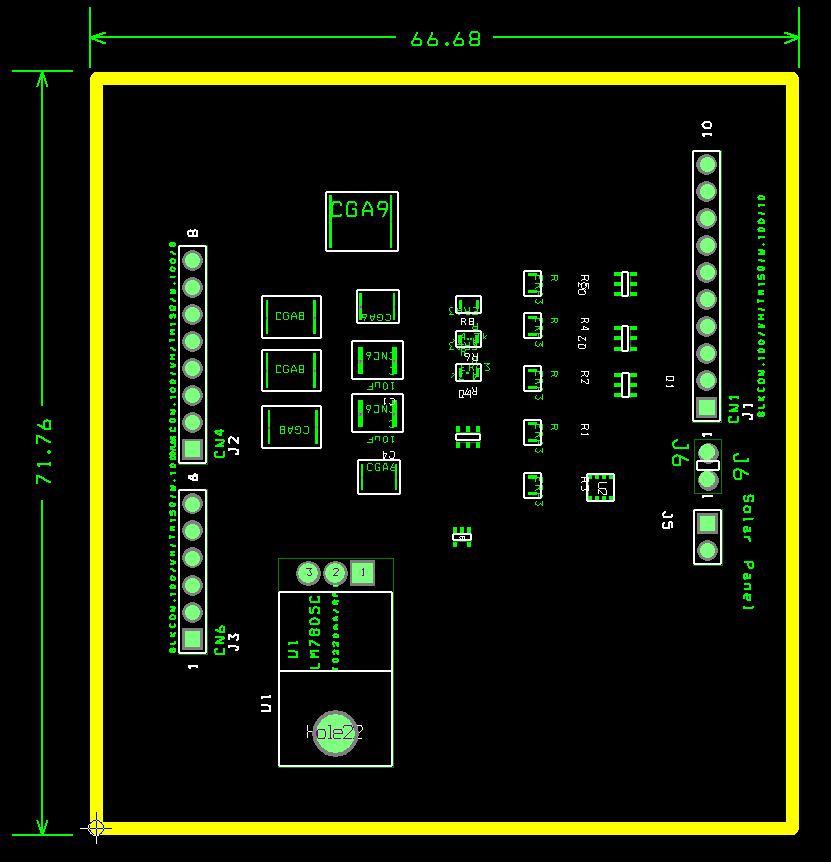
Платката е проектирана така, че да може да се включи директно към крайното LoRa устройство, правейки връзка между микроконтролерната платформа, BME280 сензора и слънчевия панел. На Фиг.4.1 е представена печатната платка, пазположението на елементите и размерите на платката.

#### **4.3.1 Опроводяване на печатната платка**

Тук е представена допълнителната за системата за охрана с видеоконтрол платка. Опроводена с писти на горния слой хоризонтално и на задния слой вертикално. На Фиг.4.2 е представено опроводеното устройство. В тази точка ще бъдат разгледани и останалите слоеве на печатната платка.

* **Горен слой (top layer)**

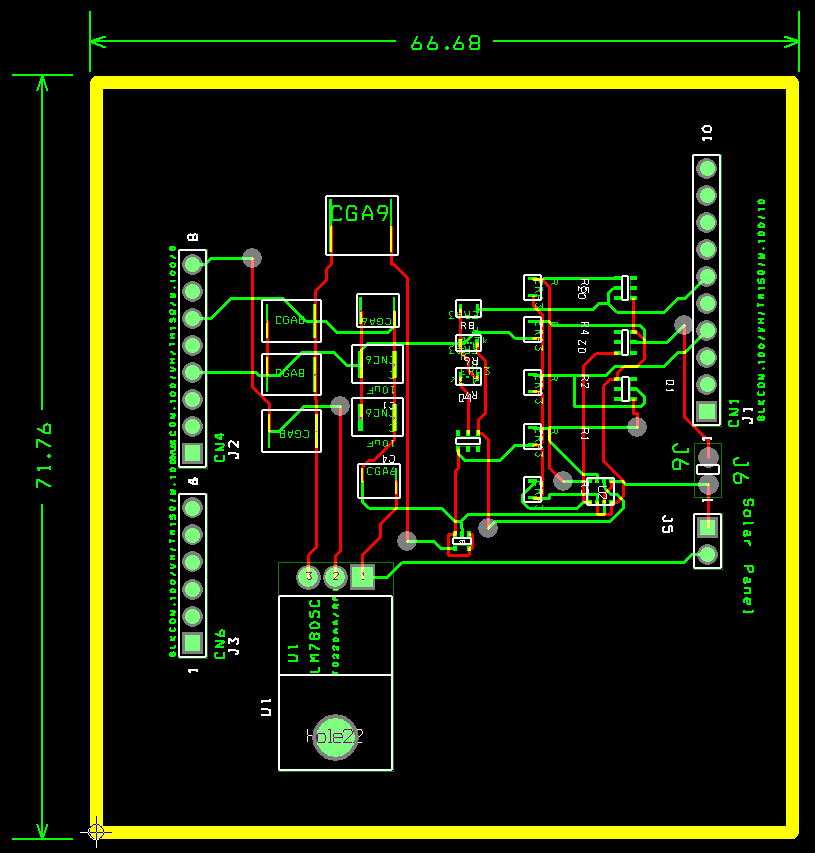
На този слой са представени как ще изглежда платката и по-ясен преглед на съединителните писти на този слой. На Фиг.4.3 е представен този слой на платката.



*Фиг.*4.1 *Разположение на елементите върху печатната платка*

* **Долен слой на платката(Bottom layer)**

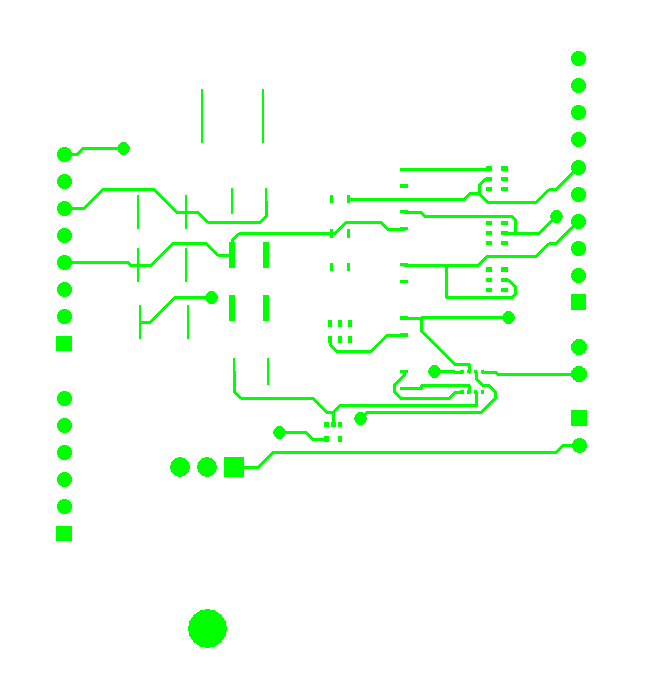
Тук е представено опроводяването върху долния слой от платката и проходните отвори на този слой. На Фиг.4.4 е представен този слой.



*Фиг.*4.2 *Опроводена допълнителна платка*

* **Документален слой за защитна маска (Solder Mask)**

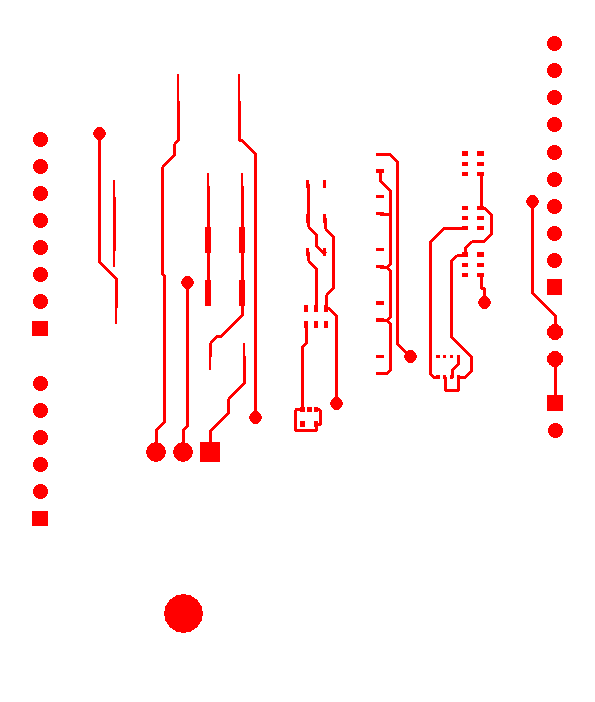
Този слой служи за защитна маска на петната от лаково покритие. Той може да отговаря както за повърхностният монтаж на опроводяване, така и за долния слой на печатната платка. На Фиг.4.5 и Фиг.4.6 са представени двата слоя на защитната маска.



*Фиг.*4.3 *Горен (Top) слой на платката*

* **Документален слой „монтажен чертеж” (Assembly Drawing)**

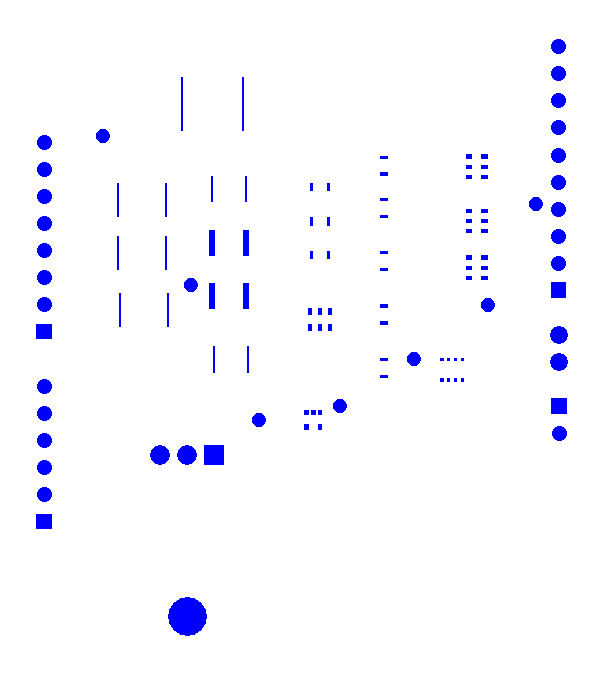
Монтажен чертеж, съдържа информация за производителя на платката, очертания на елементите, стойности, означения. На Фиг.4.7 е представен монтажния чертеж.



*Фиг.*4.4 *Заден (Bottom) слой на платката*

* **Документален слой „чертеж на отворите” (Drill Drawing)**

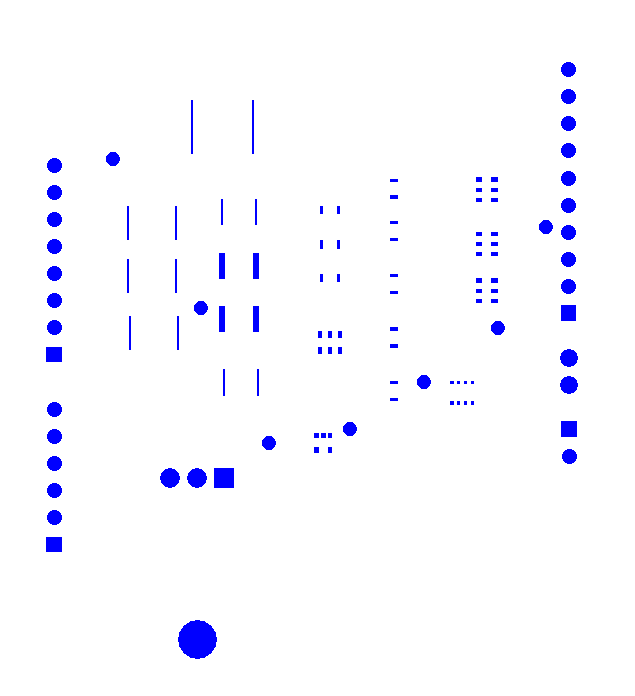
Чертеж на отворите, съдържа информация за количеството, координатите, размерите на отворите, които трябва да бъдат пробити, означени по начина, дефиниран в диаграмата на отворите (Drill Chart) с различни символи напр. +, \*, о. На Фиг.4.8 е представена таблицата за отвори, а на Фиг.4.9 слоят с чертежи на отвори.



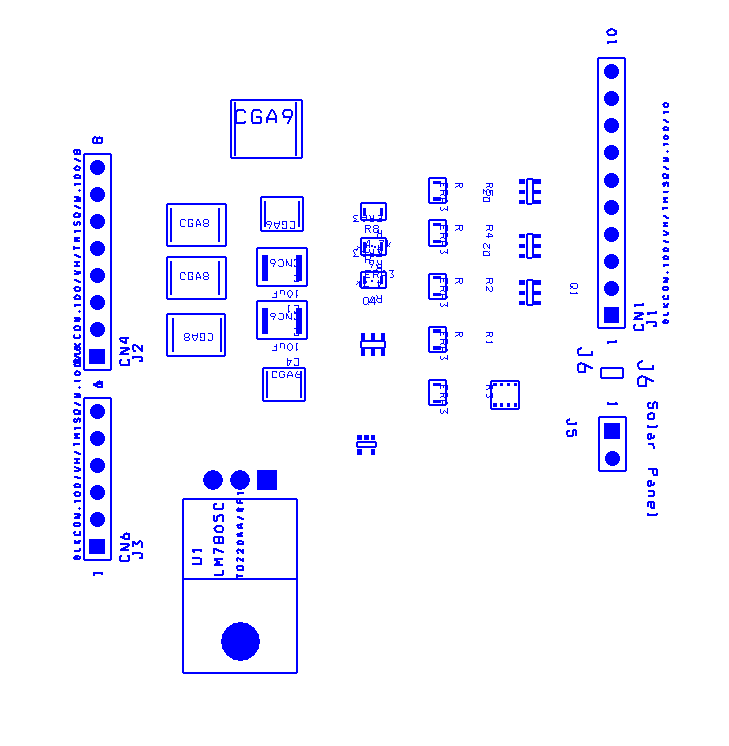
*Фиг.*4.5 *Защитна маска на top layer (SMT)*

* **Документален слой за ситопечат (Silk screen)**

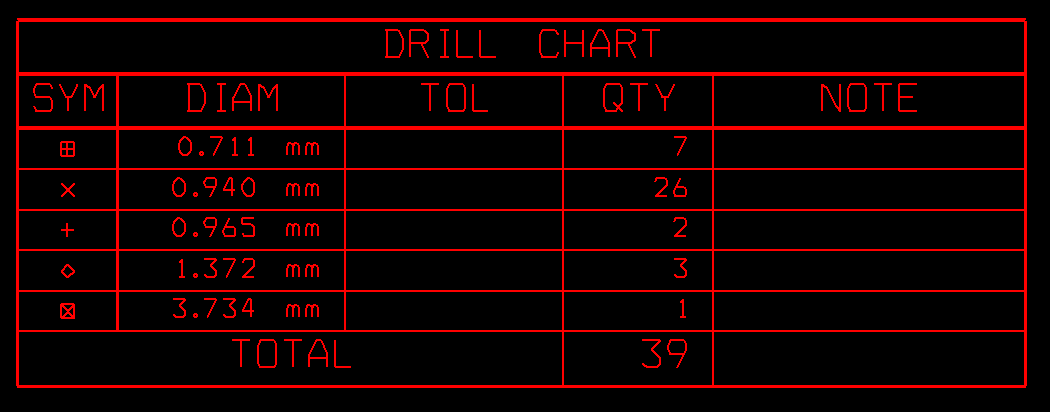
Този слой служи за надписи и очертания- ситопечат. На Фиг.4.10 е представен този слой.



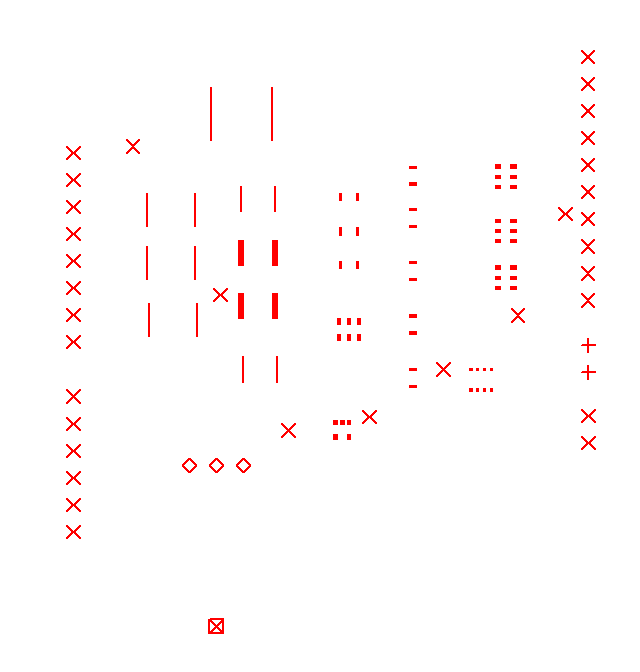
*Фиг.*4.6 *Защитна маска на bottom layer (SMB)*



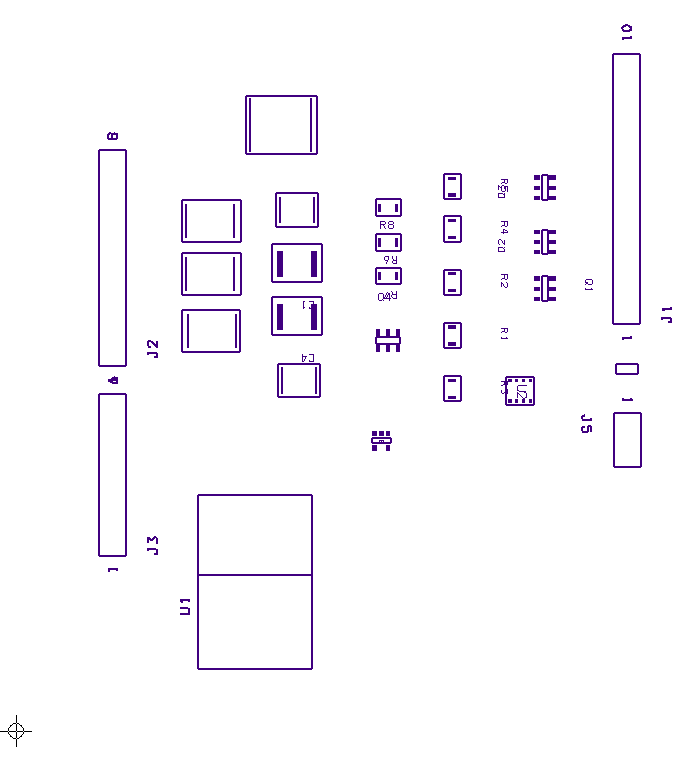
*Фиг.*4.7 *Монтажен чертеж на top layer (AST)*



*Фиг.*4.8 *Таблица на пробивните отвори*



*Фиг.*4.9 *Чертеж на отворите (DRD)*



*Фиг.*4.10 *Слой за надписи (SST)*

Глава 5

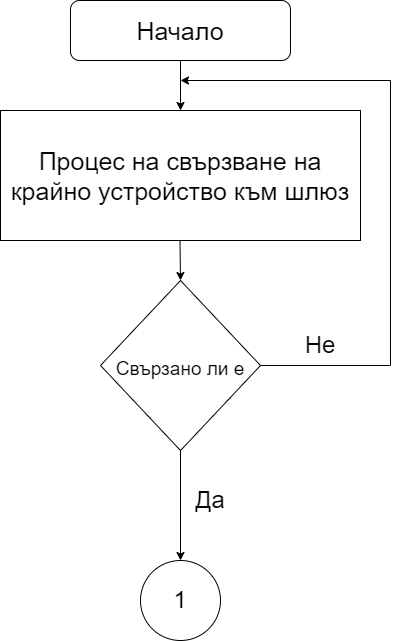
***Създаване на управляващ софтуер***

### 

### **5.1. Алгоритъм**

В тази глава ще бъде проектиран алгоритъма за работа на системата за измерване на метеорологични данни. Алгоритмичната схема ще бъде разделена на отделни блокове, които ще бъдат разгледани.

#### **5.1.1 Блок присъединяване на крайно устройство към мрежата**



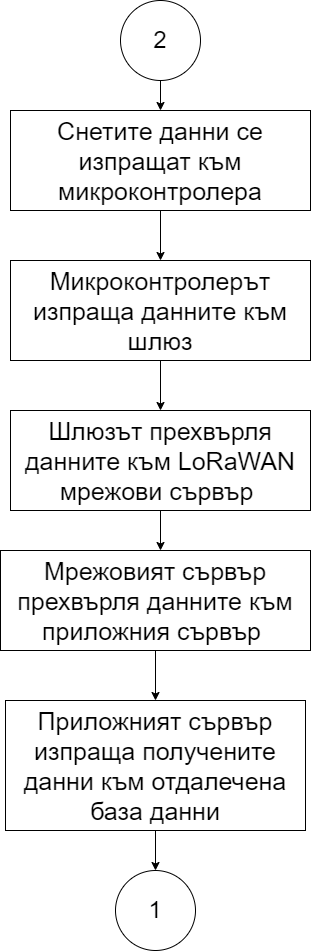
*Фиг. 5.1 Присъединяване на крайно устройство*

#### **5.1.2 Блок измерване на метеорологични показатели**



*Фиг. 5.2 Измерване на метеорологични данни*

#### **5.1.3 Обработка на снети данни**



Фиг. 5.3 Обработка на снети данни

### **5.2. Използвани библиотеки**

За свързването на крайното устройство към шлюза е използвана библиотека **"app\_lorawan.h"**

Библиотеката има следния набор от функции:

* **MX\_LoRaWAN\_Init()** - инициализация на LoRa
* **MX\_LoRaWAN\_Process()** - стартиране на LoRa планиране

За конфигуриране уникалнен идентификатор е използвана библиотека **"se-identity.h"**

| #**define** LORAWAN\_DEVICE\_EUI { 0x70, 0xB3, 0xD5, 0x7E, 0xD0, 0x04, 0xEB, 0xC6 }  #**define** STATIC\_DEVICE\_EUI 1  #**define** STATIC\_DEVICE\_ADDRESS 1  #**define** LORAWAN\_DEVICE\_ADDRESS ( uint32\_t )0x260B7488  #**define** LORAWAN\_NWK\_S\_KEY 01,FA,78,AE,9C,E1,7F,AC,1A,DC,68,8A,FC,DE,6A,18  #**define** LORAWAN\_APP\_S\_KEY 89,EE,88,BF,3C,0C,30,4A,6F,37,BD,01,8A,0D,20,04 |
| --- |
| Фиг. 5.4 |

На фиг.5.1. са показани дефинициите с които се конфигурира крайното устройство. Когато за #define STATIC\_DEVICE\_EUI стойността е равна на 1, се използва уникалният идентификатор, който потребителят задава, докато, когато е 0, автоматично се генерира стойност от микроконтролера. Подобно нещо важи и за #define STATIC\_DEVICE\_ADDRESS, когато е 1, адресът зададен от потребителя, а когато е 0, адрес автоматично се генерира от микроконтролерната платформа.

С #define LORAWAN\_DEVICE\_EUI се означава уникалният идентификатор на крайното устройство, а с #define LORAWAN\_JOIN\_EUI се означава уникалният идентификатор на сървъра за присединяване.   
 #define LORAWAN\_NWK\_S\_KEY означава мрежовия сесиен ключ, а #define LORAWAN\_APP\_S\_KEY означава приложния сесиен ключ.

За управляващ софтуер на сензора е използвана библиотека **"BME280\_STM32.h"**

Библиотеката има следния набор от функции:

* **TrimRead()** - чете параметрите за подрязване, записани в NVM ROM на устройството
* **BME280\_WakeUP()** - използва се при принудително измерване, устройството трябва да се поставя в принудителен режим всеки път, когато е необходимо измерване
* **BME280\_Measure()** - измерва температура, налягане и влажност

Глава 6

***Особеностите при създаването на работоспособен модел на проектираното устройство***

### **6.1 Конфигуриране на микроконтролер**

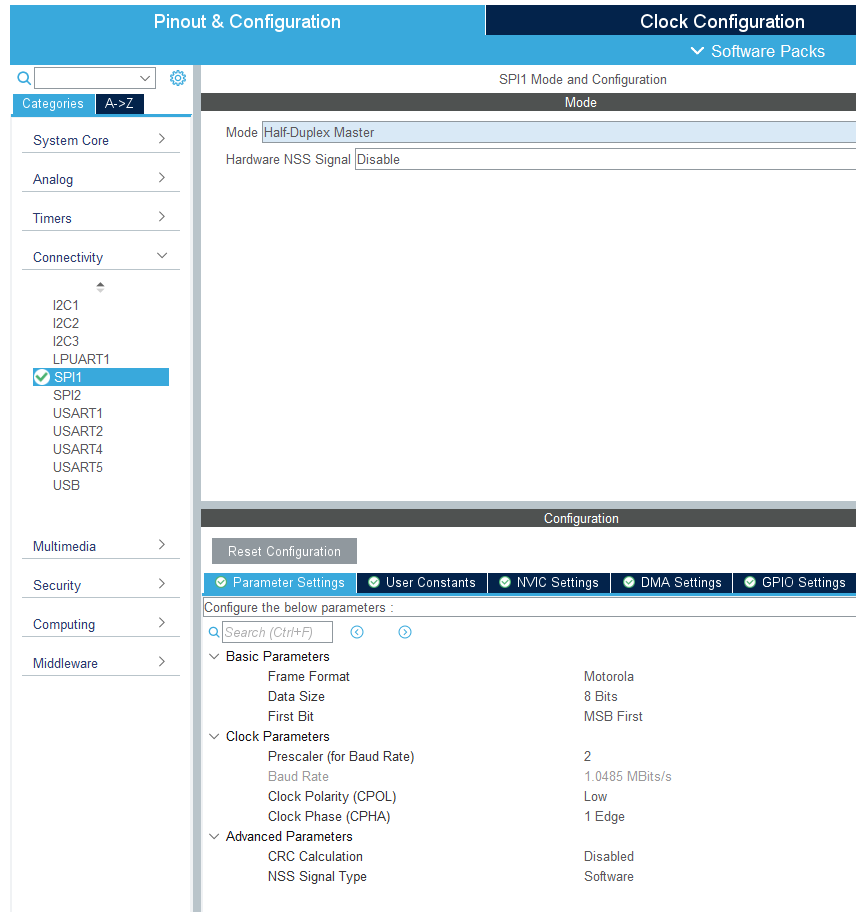
На фиг. 6.1 и 6.2 е показан процеса по конфигурация на управляващия микроконтролер. На фиг.6.1 е избран SPI начин на комуникация, а на фиг. 6.2 може да се наблюдава как са се променили изводите на микроконтролера.

### **6.2 Конфигуриране на микроконтролер**

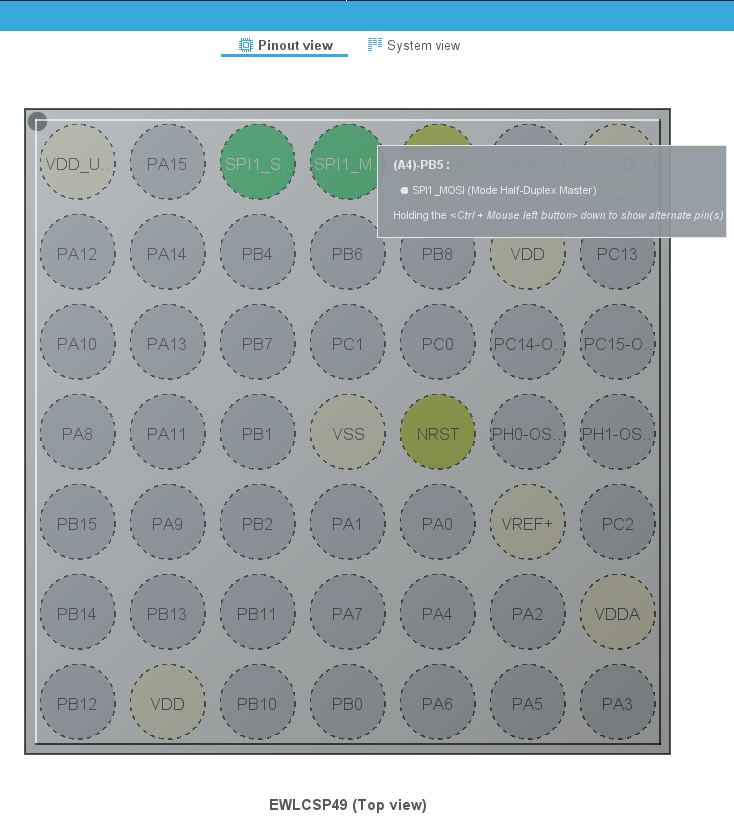
На фиг. 6.3 е показано как се създава нова апликация, а на фиг. 6.4 и 6.5 е показан процеса по добавяне на крайно устройство към апликация.

### **6.2 Конфигуриране на микроконтролер**

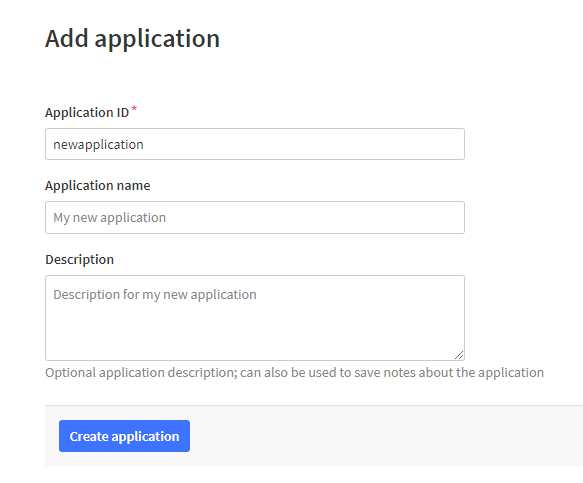
На Фиг. 6.6 е показан управляващият микроконтролер- STM32L072CZ.



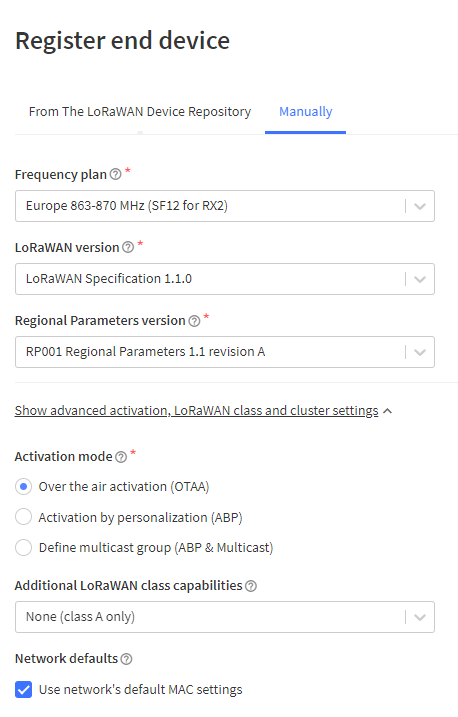
*Фиг. 6.1 Конфигуриране на микроконтролер*



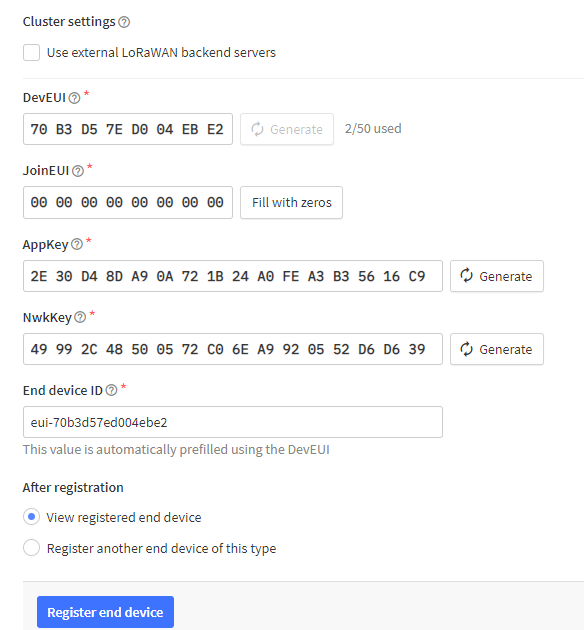
*Фиг. 6.2 Конфигуриране на микроконтролер*



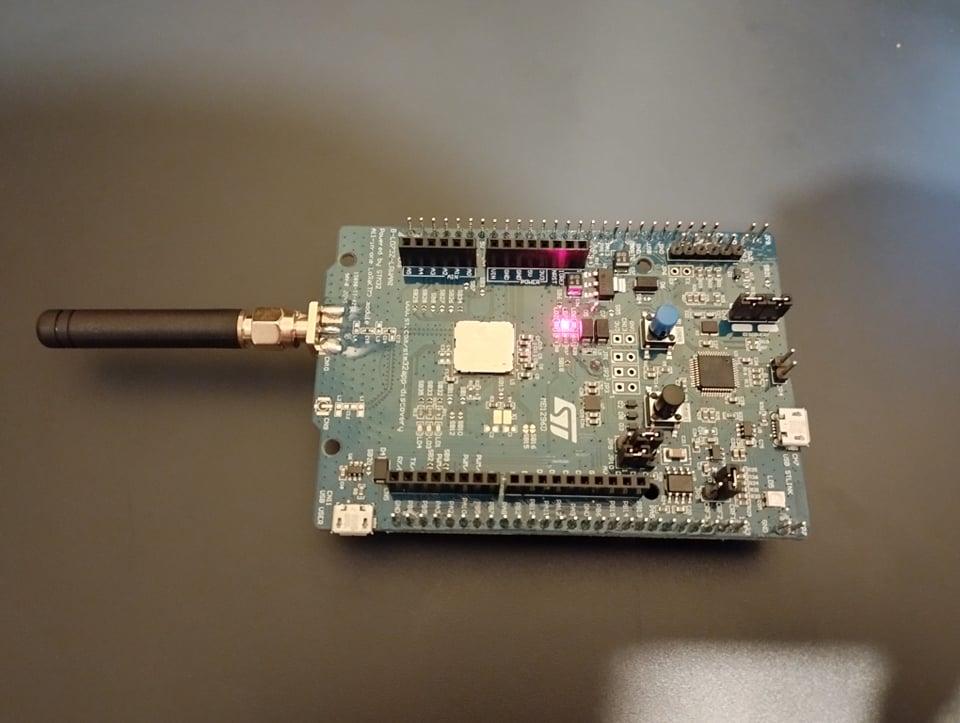
*Фиг. 6.3 Създаване на апликация*



*Фиг. 6.4 Добавяне на крайно устройство към апликация*



*Фиг. 6.5 Добавяне на крайно устройство към апликация*



*Фиг. 6.6 STM32L072CZ микроконтролер*

Заключение

За постигане на целта на настоящата дипломна работа, дефинирана в увода й, беше извършено следното: проучване, сравнение и анализ на микроконтролерни платформи и допълнителни компоненти, подходящи за разработваната система, дефиниране на основните изисквания, проектиране на логическата свързаност между периферията от устройства и допълнителни компоненти, проектирана печатна платка с възможност за допълнително захранване и изграждане на комуникация между управляващата платформа и сензора, разработване и проектиране на алгоритъм за работа на управляващия софтуер на системата. Разработването на система за измерване на метеорологични данни, основаваща се на микроконтролерна платформа, позволява гъвкавост по отношение на нейното развитие и разширение. Бъдещото развитие на системата може да включва добавяне на още сензори, за да се разшири “полезрението” на системата и добавяне на още функционалности, като прогноза за времето.

**Използвана литература:**

[1]. TMP117- <https://www.ti.com/lit/ds/symlink/tmp117.pdf?ts=1641689368334>

[2].TMP235- <https://www.ti.com/lit/ds/symlink/tmp235.pdf?ts=1641893828929&ref_url=https%253A%252F%252Fwww.ti.com%252Fsensors%252Ftemperature-sensors%252Fanalog%252Foverview.html>

[3].STM32CubeIDE- <https://www.st.com/en/development-tools/stm32cubeide.html>

4].STM32L0 Discovery kit LoRa- <https://www.st.com/en/evaluation-tools/b-l072z-lrwan1.html>

[5].Arduino Nano 33 IoT-

<https://store-usa.arduino.cc/products/arduino-nano-33-iot>

[6].Arduino Uno- <https://store.arduino.cc/products/arduino-uno-rev3/>

[7].<https://docs.espressif.com/projects/esp-idf/en/latest/esp32s2/hw-reference/esp32s2/user-guide-devkitm-1-v1.html>

[8].STM overview- <https://www.st.com/en/evaluation-tools/product-evaluation-tools.html>

[9].STM32L0CZ datasheet- <https://www.st.com/resource/en/datasheet/stm32l072cz.pdf>

[10].ESP32 datasheet- <https://www.espressif.com/sites/default/files/documentation/esp32-s2-mini-1_esp32-s2-mini-1u_datasheet_en.pdf>

[11].Arduino Nano datasheet- <https://docs.arduino.cc/resources/datasheets/ABX00027-datasheet.pdf>

[12].Bluetooth- <https://www.bluetooth.com/specifications/>

[13].Wi-Fi- <https://www.intel.com/content/www/us/en/support/articles/000005725/wireless/legacy-intel-wireless-products.html>

[14].Zigbee- <https://www.sciencedirect.com/topics/engineering/zigbee-protocol>

[15].LoRaWAN- <https://lora-alliance.org/about-lorawan/>, <https://arxiv.org/pdf/1607.08011.pdf>

[16].LoRa, LoRaWAN - <https://lora-developers.semtech.com/documentation/tech-papers-and-guides/lora-and-lorawan/>

[17]. SQL/nonSQL Data base <https://help.superhosting.bg/sql-nosql-databases.html>

[18]. P-NUCLEO-LRWAN2 <https://www.st.com/resource/en/data_brief/p-nucleo-lrwan2.pdf>

[19]. OSEPP слънчев панел <https://store.comet.bg/CatalogueFarnell/Product/813185/>

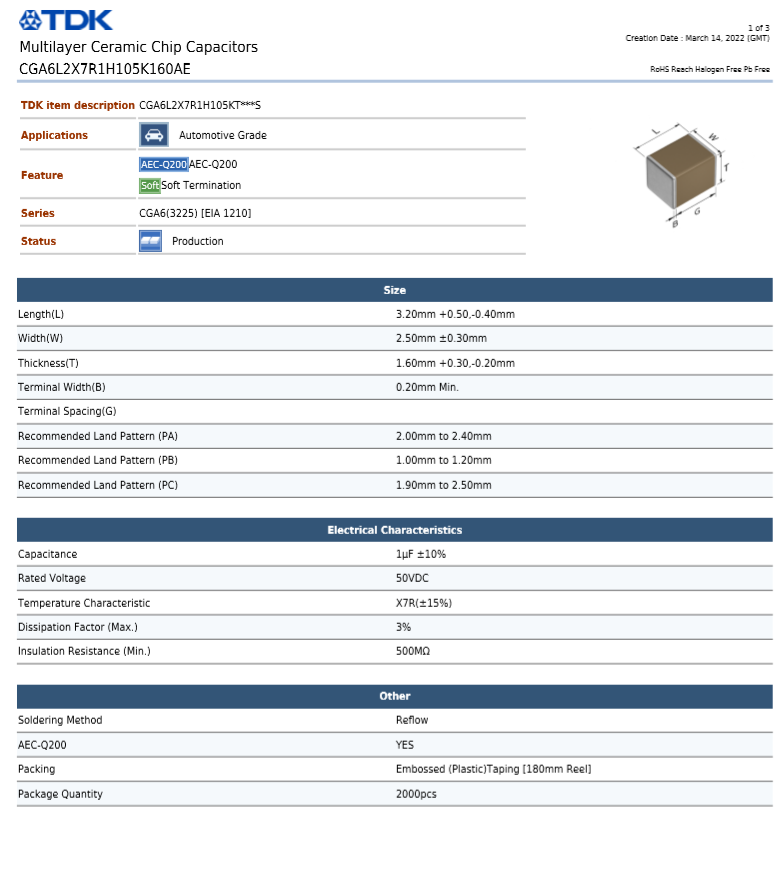
[20]. VA Bresser <https://www.bresser.de/Wetter-Zeit/Wetterstationen/NATIONAL-GEOGRAPHIC-VA-Farb-Wetterstation-4-Messbereiche-schwarz.html>

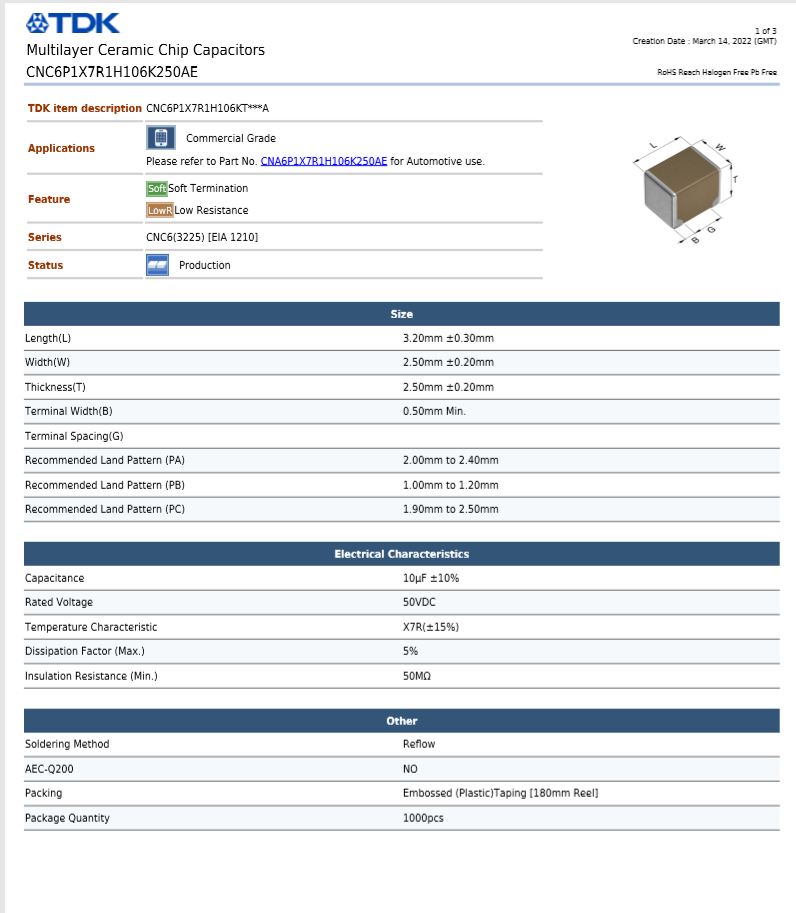
[21]. BME280 <https://erelement.com/sensors/bme280-ws?zenid=95e1lpk0afgru7r96b0pjvr8n1>

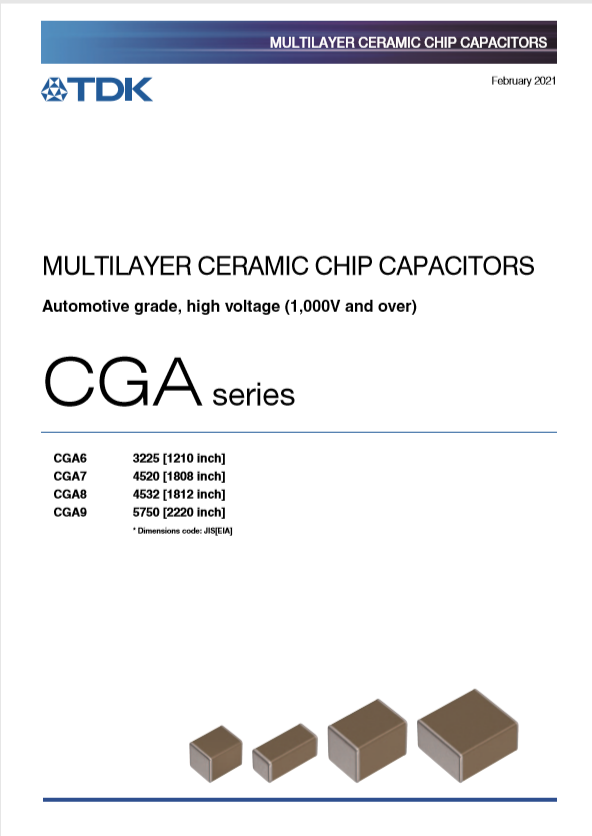
[22]. LM7805 <https://www.sparkfun.com/datasheets/Components/LM7805.pdf>

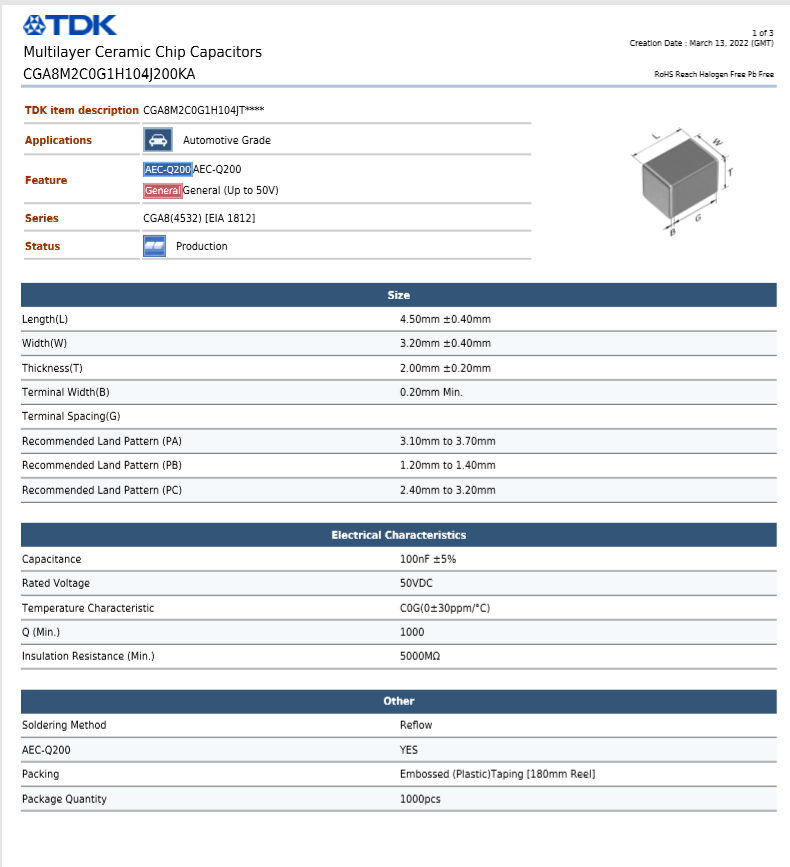
**Приложения**

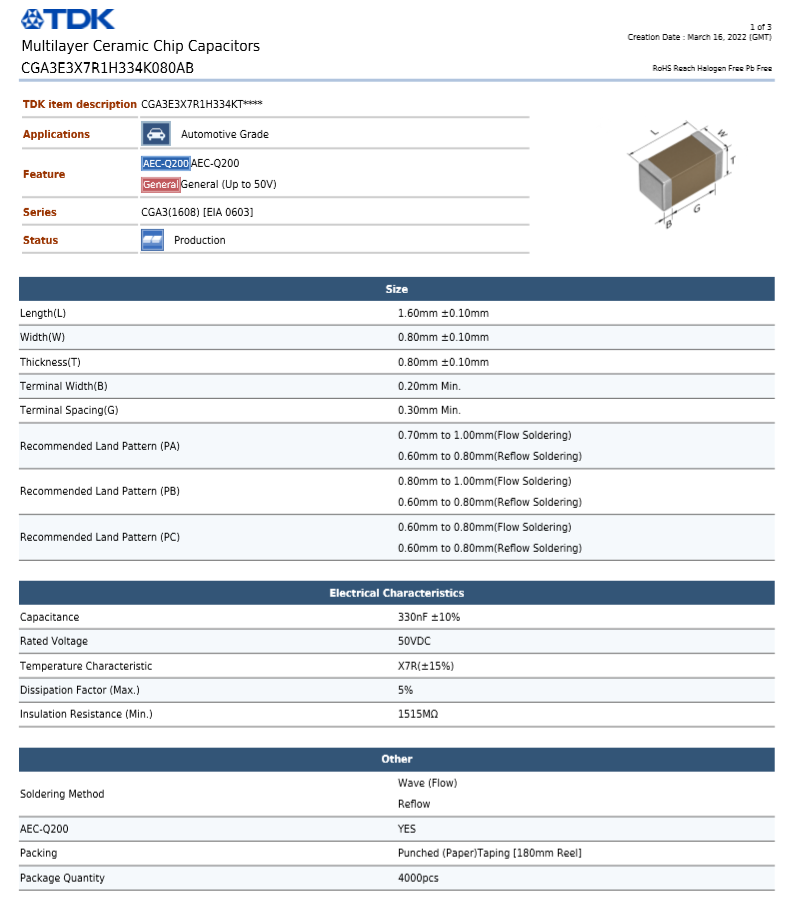
****

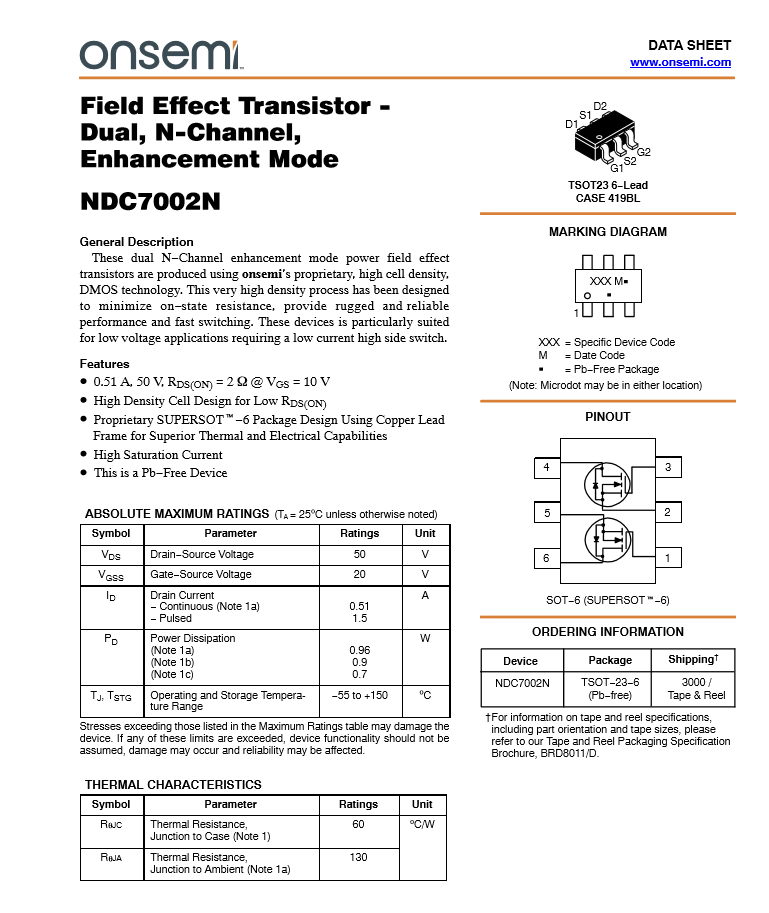
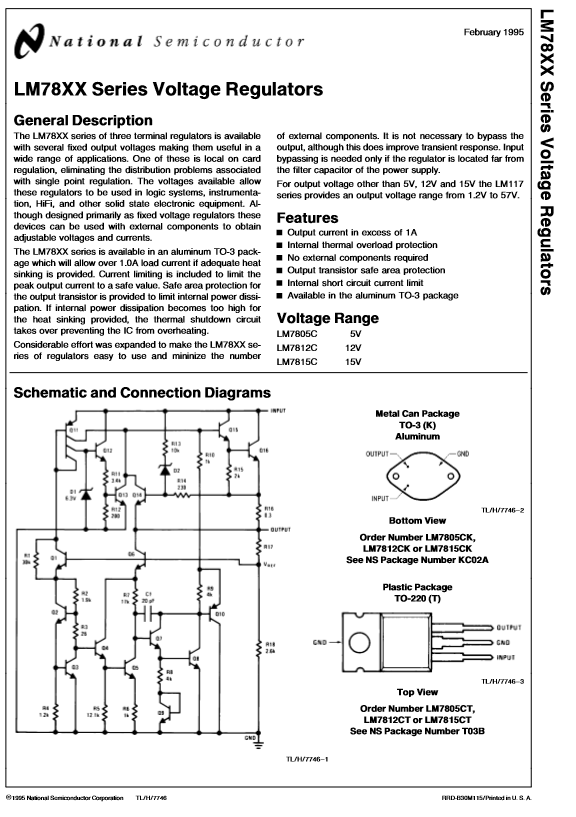
****

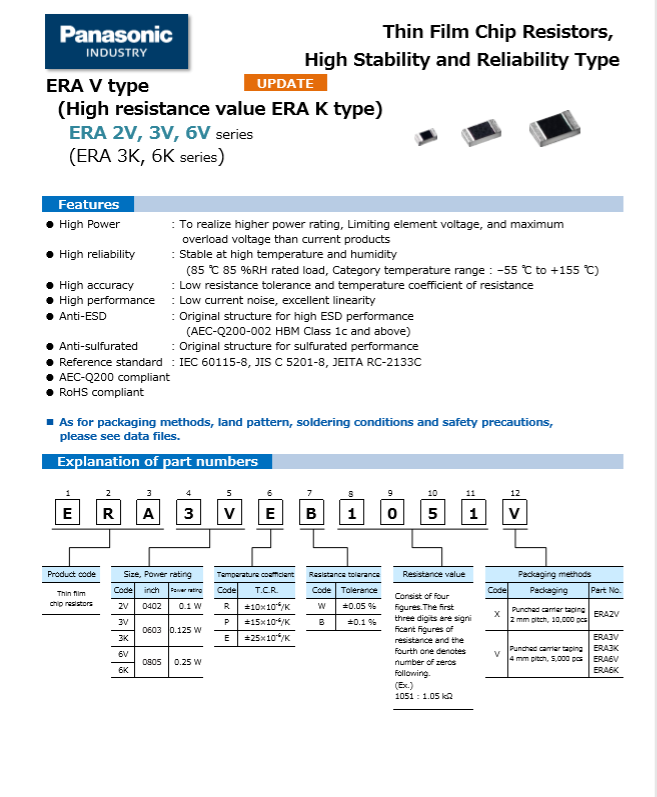
****

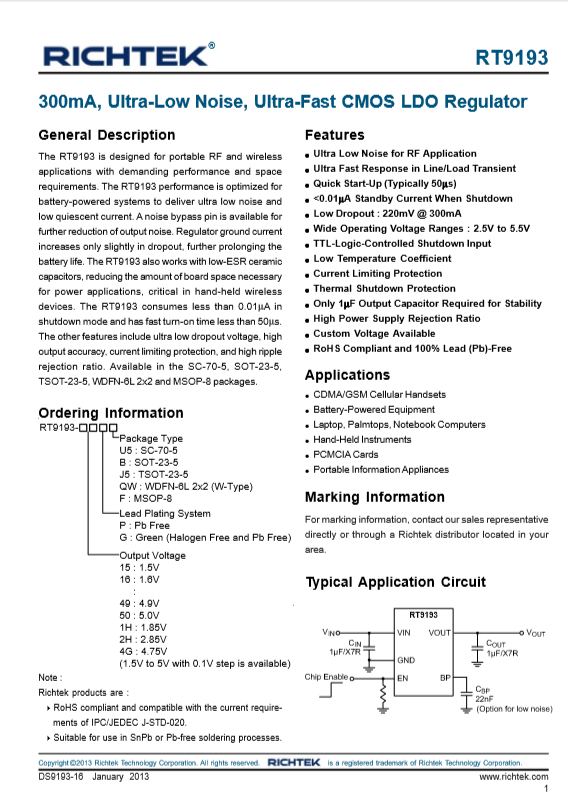
****

****

****







**Съдържание и структура на дипломната работа**

[**ТЕХНОЛОГИЧНО УЧИЛИЩЕ ЕЛЕКТРОННИ СИСТЕМИ**](#_z5foctvovih4) **0**

[**ТЕХНОЛОГИЧНО УЧИЛИЩЕ ЕЛЕКТРОННИ СИСТЕМИ**](#_1wqk8sunipnw) **1**

[1.1 . Видове сензори](#_x91z29w60y97) 2

[1.1.1 Аналогови сензори](#_y702rmtpsxgp) 2

[1.1.2. Цифрови сензори](#_hxj2laeunfom) 3

[1.2. Протокол за безжична комуникация](#_mw8zkn6bywa3) 5

[1.2.1 Wi-Fi [13]](#_iwlix1d76lsi) 5

[1.2.2 Zigbee [14]](#_w7epsemchxg7) 5

[1.2.3 Bluetooth [12]](#_salowa6j8riw) 5

[1.2.3 LoRaWAN [15]](#_uxlxkfszh6nb) 6

[1.2.4 Сравнение на протоколи за безжична комуникация](#_lly2dvdz34g6) 6

[1.2.5 LoRa и LoRaWAN[16]](#_pwgpksendz0x) 7

[1.2.5.1 LoRa[16]](#_avbklgt5paie) 7

[1.2.5.2 LoRaWAN](#_u340dhal6gj) 9

[1.3. Платформи, подходящи за реализиране на системата](#_2gcm0vkp7yov) 19

[1.3.1. Arduino [6]](#_gk20h8uumkyz) 19

[1.3.2. ESP32 [10]](#_toyshdjjt1dr) 20

[1.3.3. STМ 32 [8]](#_oxtqldchrx8o) 21

[1.3.4. Съпоставка между различните платформи](#_nnx2o8ebwb23) 22

[1.4. Шлюз [18]](#_x4z7geqvvotp) 24

[1.5. Бази данни [17]](#_651tqfkhn4u2) 24

[1.5.1. Видове бази данни](#_ul3wyp7v7hnf) 25

[1.5. Захранване](#_ubsc9up3gtmi) 26

[1.6.1. Слънчев панел [19]](#_obz4505wtrc5) 27

[Слънчевият панел има роля на допълнително захранване на системата. Ще се използва марка OSEPP, модел SC10036. На фигура 1.х е показан панелът.](#_j7got5focbug) 27

[1.7. Среда за разработка на управляващ софтуер [3]](#_pp35qtw4nsp2) 27

[1.8. Типове метеорологични станции](#_ugavhv1c4nqc) 27

[1.8.2. VA Bresser метеорологична станция [20]](#_ey53nmfh79oo) 28

[2.1. Функционални изисквания към системата](#_zievpujxclfs) 29

[2.2 Обща блокова схема и описание на отделните блокове](#_i00cj2p2k7f5) 29

[2.2.1. Описание на отделните блокове](#_9mnixlvlh2y9) 30

[2.3.1. Блок “Управление”](#_9gg83kfgdgqd) 31

[2.3.2. Блок “Измерване”](#_tkk0vnbuxmx9) 32

[2.3.3. Блок “Приемане и изпращане на снети данни”](#_afib6cx5xok) 33

[2.2.4. Блок “Захранване”](#_hh67ld2ow5mh) 35

[3.1. Избор на CAD система за проектиране](#_ya14vxsbfe5e) 36

[3.2. Избор на основни компоненти](#_xun016yw7yb7) 36

[3.2.1 Блок “BME280”](#_q85xni53z4m3) 36

[3.2.2 Блок “Voltage regulator” (Регулатор на напрежение)](#_acxrgaidhu39) 37

[3.3. Проектиране на принципна електрическа схема на системата](#_p8tik35brk) 38

[3.3.1 Йерархична структура на устройството](#_xaxofiq6flx9) 38

[3.3.2 Йерархичен блок “BME280”](#_s4dj9qullqpl) 39

[3.3.3 Йерархичен блок “Voltage Regualtor”](#_tikamvt5xz34) 41

[4.1. Проектиране на печатна платка](#_cuf0l91ox4fd) 42

[4.2. Избор на корпуси на елементите](#_k15dhfklrxrs) 42

[4.3. Проектиране на печатна платка](#_tmuue7v72c7l) 43

[4.3.1 Опроводяване на печатната платка](#_zgsf2wodcnk) 43

[5.1. Алгоритъм](#_r3r4eqraoxun) 53

[5.1.1 Блок присъединяване на крайно устройство към мрежата](#_wc8yah9p4wx1) 53

[5.1.2 Блок измерване на метеорологични показатели](#_ixwnz2fci4q7) 54

[5.1.3 Обработка на снети данни](#_3byu6qzcfuiu) 55

[5.2. Използвани библиотеки](#_dbn1oc15hjhw) 56

[6.1 Конфигуриране на микроконтролер](#_knhzwda4b8dj) 58

[6.2 Конфигуриране на микроконтролер](#_3nuwlze6ybnc) 58

[6.2 Конфигуриране на микроконтролер](#_qomv2tjb8e0i) 58