

המחלקה להנדסת תוכנה

פרויקט גמר – תשע"ו

פיתוח פרוטוקול כללי לרכיבי iot בתקשורת לוויינית
Development of protocol for iot components
(use of machine learning)

מאת: אריק לוי 303175640
גיא מימוני 200481638

| | | |
|------------------------------|--------|--------|
| מנחה אקדמי: דר' גיא לשם | אישור: | תאריך: |
| רכז הפרויקטים: מר אסף שפיינר | אישור: | תאריך: |

תוכן עניינים

| | |
|------------|-------------------------------|
| 2-3..... | תקציר |
| 3..... | מילון מונחים |
| 4..... | 1. מבוא |
| 4-5..... | 2. תיאור הבעיה |
| 4..... | א. דרישות ואפיון הבעיה |
| 5..... | ב. הבעיה מבחינת הנדסת תוכנה |
| 5-7..... | 3. תיאור הפתרון |
| 5..... | א. מהי המערכת |
| 6..... | ב. תהליכים ונתונים המערכת |
| 7..... | ג. תיאור הכלים המשמשים לפתרון |
| 7-12..... | סקר שוק |
| 13-17..... | 4. נספחים |
| 13..... | א. רשימת ספרות/ ביבלוגרפיה |
| 13-15..... | ב. תרשימים |
| 16-17..... | ג. טבלת סיכונים , טבלת דרישות |

תקציר

במסגרת לימודים אקדמאיים במכללה להנדסה, הוטלה עלינו המשימה הגדולה מכל שהיא לבצע פרויקט גמר. פרויקט בנפח של כ- 400 שעות אשר ממש את כל מה שלמדנו במסגרת התואר. החלטנו לבצע פרויקט מחקרי ואשר יש לו נפח ומשמעות לשימוש בתעשייה על מנת להתאים את עצמנו לעולם הטכנולוגי של היום ולשאוף להבין לאן נושבת הרוח בעתיד התעשייה הטכנולוגיה. הפרויקט אותו נבצע נוגע במגוון רחב של שימושים טכנולוגיים ובתקווה יגע בעתיד במגוון רחב יותר בזכותנו. במסגרת כנס טכנולוגיה בשם "Hydra" אשר אורגן על ידי הרשות לחדשנות שהיא אגף חדש במשרד לתעשייה ומסחר, בכנס השתתפו מספר מוסדות אקדמיים גדולים וגם חברות טכנולוגיה מהשורה הראשונה כגון אלביט, מוטורולה, Advancetech Wireless, ועוד, בכנס הנ"ל דיברו על חשיבות IOT ומערכות זמן אמת קריטיות אשר משתמשות בטכנולוגיה הזו, הפרויקט שלנו מסייע רבות בהתקדמות הטכנולוגיה ברכיבים אלו על מנת לפתח מערכות חכמות אשר יסייעו בגישור פערים טכנולוגיים אשר בעבר היו גדולים מדי לשיפור.

הבחירה שלנו לפרויקט מחקרי זה מכמה סיבות:

1. הנושא סקרן אותנו רבות, המחשבה לתכנת רכיב קטן שיבצע משימות גדולות ובעלות משמעות רבה כשהרכיב הזה יושב על מערכת יקרה ובעלת חשיבות. במערכות כאלו ישבו מספר רב של רכיבים שיעבדו בסנכרון מושלם וישלחו חבילות מידע בהתאם לצורכי המערכת ובהתאם לרצון הלקוחות על מנת לנתר, לעקוב ולדווח על מקרים חריגים אשר קורים במסגרת הפעילות. אנו סבורים שהפרויקט הזה ייתן לנו את הכלים הדרושים על מנת להשתלב בתעשייה כמה שיותר מהר בעיקר בגלל שאנו עובדים עם שפות וכלי פיתוח אשר הם הדבר החם הבא בתחום ההייטק כך שאנו מרוצים מאד מקיום פרויקט זה.
2. הרצון שלנו כמובן הוא ללמוד כל הזמן דברים חדשים ולהתפתח ובפרויקט הזה אנו מתעסקים עם טכנולוגיות חדשות ושפת תכנות שלא יצא לנו ללמוד ולתרגל במסגרת לימודים אקדמיים – זוהי שפת פיתוח אשר יש לה שימוש רב בתעשייה והיא נחשבת לשפה אמינה וטובה בשל הספריות הרבות שקיימות בה ונותנת למשתמש סביבת עבודה נוחה ומקצועית, כמו כן אנו עובדים עם שבב אשר לא היה לנו נגיעה אליו לפני כן במסגרת הלימודים או במסגרת מקצועית.

המשמעות בפרויקט מחקרי הוא להבין שהתחלה לא עומד לרשותנו כל הכלים והידע, ולכן אנו נדרשים לבצע מחקר רב וחיפוש נרחב ברשת על מנת למצוא את הידע והכלים הכי טובים לביצוע המשימה, במסגרת מחקר זה ניחשף רבות למה שיש לתעשייה ולתחום ההייטק להציע כך שנלמד מזה המון גם מידע שלא רלוונטי בייחוד לפרויקט שלנו, כך שזה דבר מאד חיובי.

אין ספק שהפרויקט הנ"ל הוא אתגרי ואנו שמחים על האתגר הזה ומאמינים שנצליח להגיע לתוצאות טובות לשביעות רצוננו ובהתאם לדרישות האקדמיה.

מילון מונחים

lot – "האינטרנט של הדברים", הוא רשת של חפצים פיזיים, או "דברים", המשובצים באלקטרוניקה, תוכנה וחיישנים המאפשרים תקשורת מתקדמת בין החפצים ויכולות איסוף והחלפת מידע. רשת זו צפויה להוביל לאוטומציה בתחומים רבים. האינטרנט של הדברים כולל בין השאר את תחומי "הבית החכם" ו"העיר החכמה".

כמו כן, האינטרנט של הדברים כולל טכנולוגיות שמאפשרות למשל: ניטור שתלי לב, שבבים המותקנים על חיות משק לצורכי ניטור ומעקב, כלי רכב המצוידים בחיישנים מובנים, התקני שטח המסייעים לכבאים בפעילויות חילוץ והצלה ועוד. השאיפה הגדולה של הטכנולוגיה היא שבעתיד ניתן לחבר כל התקן חשמלי לאינטרנט וליצור מערכת ורשת חכמה אשר תנתר ותפקח על כל המכשירים.

"האינטרנט של דברים מבטיח מהפך הרבה יותר נרחב, לא רק שימוש באינטרנט כלשלו רחוק אלא יכולת אוטונומית של מכשירים לתקשר אחד עם השני. דוגמה טובה להתפשטות של חיישנים חכמים היא מוני החשמל החדשים המאפשרים תקשורת דו כיוונית בין הצרכנים לרשת החשמל. בעזרת חיישנים אלו יוכלו הצרכנים לדעת כאשר ישנו מחסור בחשמל ואף לכבות באופן אוטומטי מכשירים זוללי אנרגיה בשעות בהן יש עומס על הרשת."

Application – הגורם האנושי אשר מקבל את המידע מהחיישנים אחרי שעברו את מרכז הבקרה ונשלחו אליו.

Arduino – סביבת הפיתוח אשר מאפשרת כתיבת תכניות, הדרתן והבקרתן (Flashing) לרכיבים.
אנו נעבוד בעיקר בסביבה זו על מנת לפתח את האלגוריתם לפרוטוקול.

Linkit smart 7688 duo – שבב IOT של חברת Mediatek, השבב בעל מעבד, זיכרון ראם ומטמון, כמו כן לשבב קיימים חיבורי micro-usb וניתן לחבר אותו לרשת אלחוטית.

1. מבוא

המוצר שאנו שואפים לפתח הוא פרוטוקול כללי לרכיב IOT להעברת מידע בשימוש תקשורת לוויינית.

כפי שאנו מכירים ויודעים יש מחשבים בכל מקום בעולם, גם באזורים מרוחקים ומבודדים שבהם הקליטה הסלולרית בעייתית ואין כמעט באזור חיבורי רשת חוטית ואלחוטית, אך באזורים אלו יש מכשירים חיוניים ובעלי חשיבות רבה כגון: צינורות גז, מאגרי מים, אסדות קידוח ועוד.

מה קורה כאשר יש בעיה ויש צורך להעביר את המידע כמה שיותר מהר לגורמים המוסמכים שידעו לבוא ולפתור את הבעיה?

כיום יש פרוטוקולים ורכיבים המבצעים את העבודה ומעבירים מידע למרכזי בקרה אך עלותם יקרה וכמות המידע שמועבר הוא עצום ובעל חסרון של יעילות.

רוב השימוש היום, הוא בתקשורת לוויינית שכידוע קיימת בכל נקודה בכדור הארץ אך כמובן השימוש בה יקר מאד וצורך שימוש בלוויין שיהיה זמין בכל רגע נתון כי בכל רגע עובר המידע מהרכיבים דרך הלוויין למרכזי הבקרה.

מטרתנו היא להקטין את העברת המידע למינימלית ככל שאפשר ושתועבר רק בעת הצורך לדוגמה במקרה חריג שקורה כגון עליית מד הטמפרטורה בצינור גז לרמות מסוכנות. דבר זה יחסוך בעלויות ככל שאפשר וכמובן שיעזור בייעול המערכת.

2. תיאור הבעיה

א. דרישות ואפיון הבעיה

הדרישות הן שיועבר מידע חיוני על מנת שתהיה פעילות שוטפת של המערכות ככל שאפשר ומבלי שיווצרו בעיות חדשות וחמורות. הדרישות הטכנולוגיות תמיד עולות ומחמירות ככל שמתקדמים עם השנים ומפותחות מערכות חדשות וטכנולוגיות חדשות, החברות אשר משתמשות בטכנולוגיות האלו ירצו לחסוך כמה שיותר ולייעל את המערכות, לכן המצב הקיים מצוי ולא רצוי, ודורש פתרון. כפי שהסברנו קצת במבוא, הבעיות העיקריות הן: העלות והייעול.

העלות - בשל הבעיה הגיאוגרפית אשר מונעת מלהתקין כבלים, אנטנות ומכשירי רשת אלחוטית/חוטית רוב החברות אשר מפעילות מערכות חיוניות ובעלות חשיבות משתמשות בעיקר בתקשורת לוויינית אשר כמובן יקרה יותר ומצריכה שימוש תמידי בלוויין אשר ימקד את רוב משאביו לשימוש זה, המידע כמובן עובר בכמויות רבות אשר מצריכות שימוש בשרתים רבים בעלי כוח אחסון ועיבוד רב ובנוסף מרכזי בקרה אשר יתחזקו את הפעילות ויעבירו את המידע מסוכן ויותר קונקרטי לגורמים המוסמכים. עלות רוחב פס בלוויין היא עצומה כמו כן אחזקת שרתים רבים וחזקים בעלי יכולת עיבוד משמעותית היא לא זולה.

הייעול – העברת המידע עוברת בצורה כמעט תמידית וצורכת שימוש ברוחב פס של תקשורת לוויינית בצורה כמעט רצופה, המידע שמועבר הוא מידע רב אשר מתוכו מעט ממנו רלוונטי ונחוץ לשם העבודה השוטפת, רוב הזמן המידע הוא מידע שגרתי אשר מתאר את הפעילות השוטפת של החיישנים ושל המערכות אשר עליהן מותקנים החיישנים. כפי שרואים פעילות זו לא יעילה ולא עובדת בצורה מינימלית.

ב. הבעיה מבחינת הנדסת תוכנה

הבעיה הראשונה שאנו צריכים להתמודד אתה עם עבודה עם ממשק וסביבות עבודה שלא היה לנו נגיעה איתם עד כה. הבעיה השנייה שלנו היא לזהות את שינוי מצב המערכת במקרה והתבצע מקרה קצה כלשהו, לתפוס את השינוי מצב ולעבד אותו לחבילת מידע.

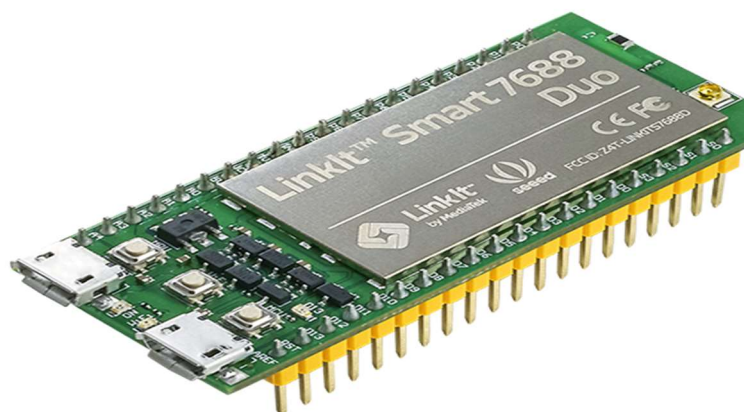
3. תיאור הפתרון

השאיפה שלנו הוא ליצור פרוטוקול כללי לשבב IOT על מנת לפתור את אותם בעיות אלו. הרצון שלנו שהשבב יעביר מידע **מועט ככל שאפשר ורק במקרה הצורך** כמו לדוגמה כאשר קורה משהו, ולא בצורה תמידית ושוטפת.

א. מהי המערכת

השבב אמור להעביר כל הזמן ביט אחד כתשובה לכך שהמצב תקין והכל עובד בצורה טובה במערכות המרוחקות, וכאשר יש בעיה או שינוי מסוים השבב (חיישן) יעביר עוד ביטים למרכזי הבקרה כדי להמחיש שישנה בעיה במערכת, כמובן שהמידע עצמו שיועבר ייתן לגורמים המוסמכים את התמונה הרחבה על מה היא הבעיה.

אנו נשתמש בשבב IOT בשם Linkit Smart 7688 Duo של חברת מדיה טק (ראה תמונה מתחת).



אנחנו מרגישים שככל שיוגדל השימוש בשבב הנ"ל למשל בזכות הפרויקט שלנו כך גם הביקוש יעלה ובזכות הדרישה למוצר, העלות שלו תרד, דבר שיחסוך כסף רב לחברות ולתעשייה. הרכיב בעל מפרט לא רע בשל גודלו:

- זיכרון של 128 מגה בייט,
- מעבד MIPS24KEc – 32 bits, 580 MHz.
- אפשרות להרחבת זיכרון עד 512 מגה בייט – Micro SD Slot.
- חיבור USB 2.0.
- תומך ב – Wi-Fi.

בחרנו בו בעיקר בגלל שהוא מספר סיבות אותם נפרט בהמשך. המטרה שלנו להמחיש את הפרוטוקול הוא על מקרה ספציפי של שינוי מצב, המחשבות שלנו ללכת על נושא של עיבוד תמונה ולהראות את פעולת הפרוטוקול על מקרה של תמונה משתנה. השאיפה שלנו גם היא שבסופו של דבר נרכיב על הפרוטוקול אלמנטים של Matching learning שיגרמו לפרוטוקול ולשבב לדעת וללמוד מעצמו מתי להעביר את המידע ואיך. שפת התכנות התומכת בשבב הזה ועליה נפתח את הפרוטוקול היא פיתון, וכיוון שלא למדנו את השפה הזו במסגרת הלימודים אנו נלמד אותה בצורה עצמאית וזה כמובן יוסיף לאתגר שלנו לממש את הפרויקט הגדול הזה.

ב. תהליכים ונתוני המערכת

המערכת תכלול 2 מצבי עבודה עיקריים:

מצב תקין - מצב זה יהיה באופן שוטף כאשר אין לחיישן משהו חדש שהוא זיהה, מצב זה אומר שמצב המערכת תקין והחיישן לא מזהה משהו מיוחד אשר מצריך שינוי מצב ושליחת מידע נוסף, במצב זה ישלח בצורה שוטפת או לסירוגין (לדוגמה: פעם בכמה דקות) ביט בודד ללא מידע נוסף אשר יהווה הוכחה למרכז בקרה שהכל תקין במערכת.

מצב אירוע חריג - מצב זה יקרה כאשר קרה אירוע חריג (לדוגמה: עליית טמפרטורה חריגה מהנורמה), מצב זה יצריך מעבר של ביטי מידע נוספים למרכזי הבקרה באופן מידי וללא דיחוי, כמובן שמצב זה ושליחת מידע נוסף יעיד שמצב המערכת עליה מותקן החיישן – לא תקין, הרצון הוא שתותקן על החיישן או על רשת החיישנים אלמנטים של Matching Learning אשר תדע לתפעל את האירוע ולקבוע האם המצב הוא חריג מאד או האם הוא בר תיקון ברמת המערכת (למשל: טמפרטורה חריגה במעט מהנורמה אך יכולה עדיין להיות נסבלת מבחינת המערכת), דבר זה לא יצריך שליחת המידע למרכז הבקרה.

הפתרון יהיה שמרכז הבקרה יקבל מידע אחת לכמה זמן ו/או במקרה חריג יקבל מידע נוסף על שינוי המצב, השאיפה היא שהאלגוריתם ילמד את עצמו לתפעל ולפתור את הבעיות כך שהמערכת תמשיך לשלוח ביט בודד להראות שהמצב תקין, פתרון זה כמובן באופן ישיר יחסוך בעלויות אחסון ותשתיות רחב פס לשימוש במערכת תקשורת לוויינית אשר יקרה במיוחד וכמובן שרתים לאחסון המידע.

ג. תיאור הכלים המשמשים לפתרון

בפרויקט נשתמש בסביבת הפיתוח Arduino שמאפשרת כתיבת תכניות וסביבת עבודה לשבבי חישה של IOT, אנו נעבוד עם שפת התכנות python אשר קיימות לה ספריות רבות והיא נחשבת לאחת השפות הפופולריות לפיתוח.

הפרויקט ינוהל לחלוטין על ידי מערכת ה GitHub אשר תספק לנו:

- מאגר הקוד.
- מערכת משימות ומטלות.
- מצב הפרויקט.
- מסמכים ומידע נוסף שרלוונטי לפרויקט.

כמו כן אנו ננהל יומן אירועים של מהלך הפרויקט מתחילתו ועד סופו באתר [Trello](https://trello.com/).




סקר שוק

המחשבה העיקרית בפרויקט הזה הוא לחסוך בעלויות, למשל בשימוש בשבב זול אך שיכיל את המשאבים אותם אנו צריכים על מנת שפרויקט זה יצליח כגון זיכרון אחסון, אפשרויות חיבור למערכות אחרות (לדוגמה: מחשב, רשת אלחוטית ועוד..), ומעבד פנימי שיוכל להריץ את הפרוטוקול ללא תקלות או קריסות.

בנוסף אנו מפתחים פרוטוקול שמטרתו לצמצם התקשרות עם מתקן ספציפי בשליטה מרחוק למשל באמצעות רוחב פס לווייני וכמובן זה עוזר לחסוך בעלויות שרתי אחסון מידע במרכזי הבהרה. בעקבות סיבות אלו החלטתנו לעשות סקר שוק על שני דברים שונים: על שבבי IOT אשר יוכלו לספק לנו סביבת עבודה ופיתוח ועל פרוטוקולים דומים שבוצעו בעבר בתחום IOT בנושא העברת מידע מינימלית.



א. שבבי IOT

תחום ה IOT הוא נרחב היום בתעשייה ואף יהיה נרחב יותר בעתיד, Cisco מעריכים שב-2020 יהיו כ-20 מיליארד מכשירים IOT בשימוש, ולכן אף עכשיו יש מגוון של חברות שמפתחות שבבים מורכבים שבגדול אפשר לקרוא להם מחשבים קטנים כי הם מכילים את כל מה שיש במחשב כגון: מעבד, זיכרון ram, זיכרון אחסון, חיבורים חיצוניים: usb, type-c ועוד. אנו נרצה לעשות השוואה בין שבבים שונים שניתנים היום לקנייה ובעזרת פרמטרים אלו נעיד למה בחרנו דווקא בשבב Linkit כדי שישמש קרקע לסביבת פיתוח.

| | <u>UDOO Dual Basic</u> | <u>LinkIt Smart 7688 duo</u> | <u>Raspberry Pi Model A+</u> |
|--------------------------|---|--|---|
| |  |  |  |
| CPU | ARM Cortex-A9 (32-bit) 1000MHz dual-core | MIPS24KEc (32-bit) 580MHz single core | RISC ARM11 (32-bit) 700MHz single core |
| RAM size | 1GB | 128MB | 256MB |
| Internal storage | X | 32MB | X |
| SD card | MicroSD card slot Max. size: 32GB | MicroSD card slot Max. size: 512GB | MicroSD card slot Max. size: 32GB |
| USB host | 2 | 1 | 1 |
| Ethernet | X | 1 port | X |
| Wi-Fi | X | 1T1R 802.11 b/g/n (2.4G) | X |
| Power consumption | 1.2A - 2A | 300mA - 1A | 100mA - 600mA |
| Price | 99\$ | 16\$ | 20\$ |

בחרנו לעשות השוואה בין 3 שבבים, לכל אחד מהם יש את היתרונות שלו, לדוגמה לשבב UDDO יש זיכרון Ram יותר גדול, אך בהתאם לצרכים שלנו שהם זיכרון אחסון מובנה, אפשרות הרחבת זיכרון גדולה, חיבוריות ל ethernet, אך היה גורם אחד שהכריע בצורה הכי משמעותית והוא המחיר, אמנם הבדל בין Linkit ל- Raspberry הוא כ- 4 דולרים בלבד אך כאשר מרכיבים מספר רב של שבבים על מערכת הפעם במחירי עלות עולה בצורה משמעותית עד כדי הבדלים של אלפי דולרים למערכת שבבים, ולכן בשלב זה ה- Linkit מנצח.

העמקנו את החיפוש ברשת בניסיון לחפש שבב אחר שאולי מחירו יהיה שווה או יותר זול מהשבב שלנו ועדיין לא יגרע מאיכותו מבחינת המפרט, מצאנו עוד שבב בשם **Orange Pi PC** ולהלן ההשוואה.

| | LinkIt Smart 7688 duo | Orange Pi PC |
|--------------------------|---|---|
| |  |  |
| CPU | MIPS24KEc (32-bit) 580MHz single core | ARM Cortex-A7 (32-bit) 1.6GHz quad core |
| RAM size | 128MB | 1GB |
| Internal storage | 32MB | X |
| SD card | MicroSD card slot Max. size: 512GB | MicroSD card slot Max. size: 64GB |
| USB host | 1 | 3 |
| Ethernet | 1 port | 1 port |
| Wi-Fi | 1T1R 802.11 b/g/n (2.4G) | X |
| Power consumption | 300mA - 1A | 700mA - 2A |
| Price | 16\$ | 15\$ |

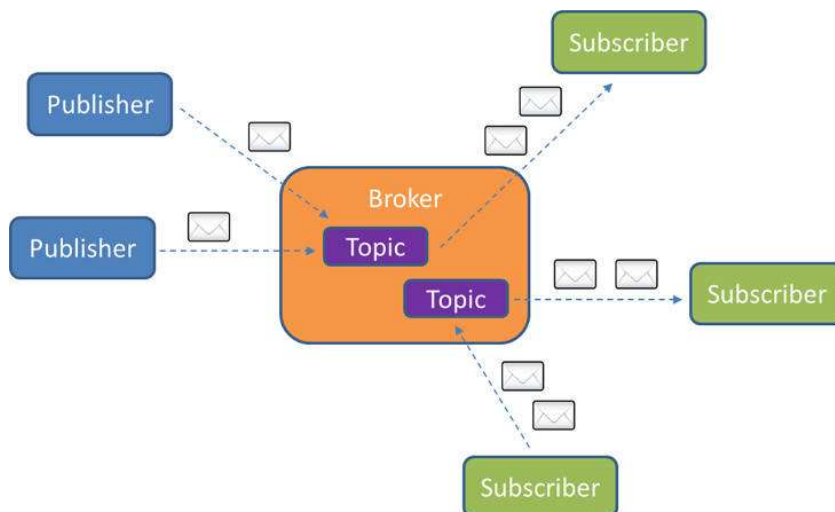
כפי שניתן לראות כאן קריטריון המחיר נוטה לשבב האורנג' ויש לו עוד קריטריונים שבהם הוא מנצח אך אין אלה הקריטריונים החשובים, לפי הצרכים שאנו מבקשים מהשבב ותפוקתו, חשוב לנו שהוא לא יצרוך חשמל רב במיוחד שיורכבו על מערכת מספר רב של שבבים, כמו כן לשבב ה-LinkIt יש זיכרון Ram מובנה בעוד לשבב השני אין, את הפרוטוקול אנו מריצים על השבב אנו נרצה שיהיה לו זיכרון אחסון ולכן זה הקריטריון הכי חשוב, בנוסף קריטריון אחרון שיש לשבב ה-LinkIt הוא חיבור ל Wi-Fi שזה יכול לעזור בחיבוריות בין חיישנים שונים מרחוק בניגוד לחיבור פיזי ביניהם, כמו כן זה יעזור לנו בפיתוח. ולכן הבחירה שלנו הייתה נחרצת ובחרנו בשבב ה-Linkit smart 7688 duo.

ב. פרוטוקולי IOT להעברת מידע

1. MQTT - פרוטוקול חיבוריות ממכונה למכונה (M2M) כמו כן הוא פועל גם ברכיבי IOT, פרוטוקול זה קל מבחינת זיכרון ויכולת עיבוד כך שאפשר להפעיל אותו על רכיבים וחיישנים קטנים, בנוסף הוא אידיאלי לשימוש ברשת חיישנים בחיבור מרחוק ע"י לווין.

הפרוטוקול פותח ב-1999 על ידי אנדי סטנפורד-קלארק וארלן ניפר. ב-2013 חברת IBM הגישה ל OASIS (ארגון הקובע סטנדרטים בתחום התוכנה) את MQTT v3.1 ומאז היא הפכה לסטנדרט עולמי, לפרוטוקול הזה מגוון שימושים בתעשייה וביניהם פייסבוק מסנג'ר אשר משתמש בפרוטוקול זה. המאפיין העיקרי שלו הוא ללא איבוד נתונים ולכן הוא יושב מעל שכבת TCP ובכך מאפשר פשטות ושליחת נתונים אמינה.

דפוס העבודה ב MQTT הוא של publish/subscribe, כל מכשיר יכול לפרסם מידע בנושא מסוים לכל שאר המכשירים שעשו מנוי לנושא זה, ולהפך, הוא יכול לעשות מנוי לנושא מסוים. כל המידע הזה עובר דרך המתווך שמקושר לכל המכשירים, וזוהי כל העבודה שלו.



יתרונות:

- קל מבחינת זיכרון ויכולת עיבוד
- מאפשר הפעלה מרחוק על רשת חיישנים ע"י חיבור לווין
- העברת מידע מהירה מעל TCP

חסרונות:

- לא עובד בזמן אמת, ולרוב זמן אמת נחשב אצלו כמה שניות.
- מסובך להפעלה, קשה להתקנה.

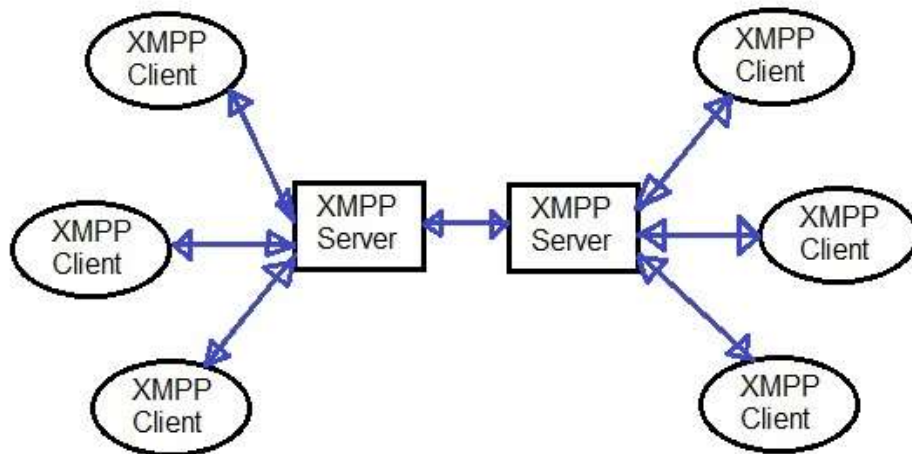
הפרוטוקול הזה דומה במהותו למה שאנו רוצים להשיג מהפרוטוקול שלנו, קל זיכרון, מאפשר העברת מידע בצורה פשוטה וניתן להפעיל אותו מרחוק.

2. XMPP – פרוטוקול תקשורת והעברת נתונים, מאפשר החלפת מסרים כמעט בזמן אמת, במקור נקרא "Jabber", והוא פותח על ידי קהילת קוד-פתוח ששמה ג'אבר בשנת 1999.

הוא תוכנן להיות בר הרחבה, והשתמש בו בין היתר להעברת קבצים, וידאו, משחקים וכמובן IOT ביישומים כגון רשת חכמה. בניגוד לפרוטוקולים אחרים בתחום, XMPP תוכנן כפרוטוקול פתוח בעל הפצה חופשית ולכן כל אחד יכול להשתמש בקוד ולבנות לו XMPP מותאם אישית.

מנגנון הפרוטוקול עובד בצורה יותר יעילה בעיקר בזה שהוא עובד על ידי דחיפת ההודעות בניגוד לפרוטוקולים דומים כדוגמת Http שמשתמש בבקשות ותגובות, בהקשר של IOT, הוא מאפשר דרך קלה להגיע להתקן וליצור לו כתובת ברשת, זה מאד שימושי אם הנתונים עוברים מרחוק, בדרך כלל בין 2 נקודות לא קשורות.

הוא לא נועד להיות מהיר ולרוב השימוש שלו הוא במשיכות (pull) מידע במקרה הצורך, במונחים אנושיים זמן אמת שלו יהיה כמה שניות, הוא מספק דרך מצוינת לחבר מכשירי חשמל ביתיים (כגון: תרמוסטט) לשרת אינטרנט כך שתוכל לגשת אליו מהפלאפון שלך.



יתרונות:

- פרוטוקול פתוח, בר הרחבה.
- יכולת חיבור מכשירי חשמל IOT – לשרת אינטרנט.
- מחבר בין 2 נקודות בצורה לא ישירה, 2 הנקודות לא חייבות להכיר באופן ישיר אחד את השני.

חסרונות:

- איטי ולא עובד בזמן אמת.
- לא עובד ללא תלוי שרת XMPP.

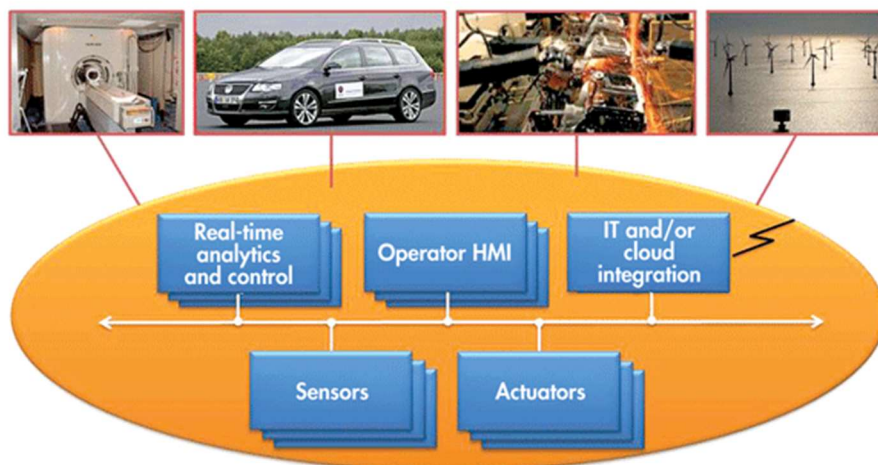
אנו נחקור מעט על הפרוטוקול הזה כי יש בו פיצ'רים שיכולים לעזור לנו בפיתוח הפרוטוקול שלנו.

3. DDS - בניגוד ל MQTT ול- XMPP, שירות הפצת הנתונים (DDS) מתמקד במכשירים שמשתמשים באופן ישיר בזיכרון המכשיר, המטרה העיקרית של הפרוטוקול הזה הוא לחבר מכשירים מסוימים לאחרים, בניגוד לאחרים DDS יכול להעביר מיליוני הודעות לשנייה בצורה יעילה ואמינה לכמה מכשירים בו זמנית.

מנגנון הפרוטוקול עובד בצורת מפרסם/מנוי, היות והוא מציע שירות למלא מכשירים עם הרבה הודעות בו זמנית, חיבור TCP לא ישים פה כי הוא מאפשר בדרך כלל חיבור מוגבל בין 2 נקודות ולכן ב DDS יש שימוש ב quality of service (בקרת איכות מפורטת), אשר דוגלת בשידור בו זמנית לכמה מכשירים (multicast), ניתנת להגדרה ותמיכה יתירה.

פרוטוקול זה מציע דרך עוצמתית לסנן איזה מידע ילך ולאן, וה- "לאן" יכול להיות אלפי מכשירים בו זמנית, הוא מאד דומה למעין אוטובוס-מידע, אוטובוס המידע שולט על כניסת המידע ויש לו דימוי של מסד נתונים כי במסד נתונים יש שליטה על הרשאות הגישה אל המידע (data) המאוחסן, חלק מהמכשירים קטנים ולכן יש גרסאות קלות של DDS שרצות עליהן בצורה מוגבלת.

מערכות משולבות בעלות ביצועים גבוהים משתמשות ב- DDS, זוהי הטכנולוגיה היחידה שמספקת את הגמישות, האמינות והמהירות הדרושים לבניית יישומים מורכבים בזמן אמת. דוגמה לכאלו: מערכות צבאיות, חוות רוח, מערכות אינטגרציה של בתי חולים, מכשירי הדמיה רפואית ועוד.



יתרונות:

- מהיר ומשדר בזמן אמת (מיקרו-שניה).
- יכול לתקשר עם אלפי מכשירים בו זמנית.
- מסוגל לשדר מיליוני הודעות בשנייה.

חסרונות:

- לא קל לשימוש, דורש ידע מעמיק ברשתות אינטרנט ומודל השכבות.
- פחות אמיין כי הוא לא משתמש בחיבור TCP.

4. נספחים

א. רשימת ספרות \ ביבליוגרפיה

Internet of Things (IoT): A vision, architectural elements, and future directions

<https://arxiv.org/ftp/arxiv/papers/1207/1207.0203.pdf>

IoT gateway: Bridging wireless sensor networks into internet of things

<http://yzhang.org/courses/20172018tcpip/papers2read/0301.pdf>

The Internet of Things (IoT): Applications, investments, and challenges for enterprises

https://s3.amazonaws.com/academia.edu.documents/41516575/The_internet_of_things.pdf?AWSAccessKeyId=AKIAIWOWYYGZ2Y53UL3A&Expires=1510592709&Signature=wdPAewiQ%2FCUyKUI8MLk8gcmo7tA%3D&response-content-disposition=inline%3B%20filename%3DThe_Internet_of_Things_IoT_Applications.pdf

IoT Standards & Protocols Guide

<https://www.postscapes.com/internet-of-things-protocols>

Scikit-learn: Machine learning in Python

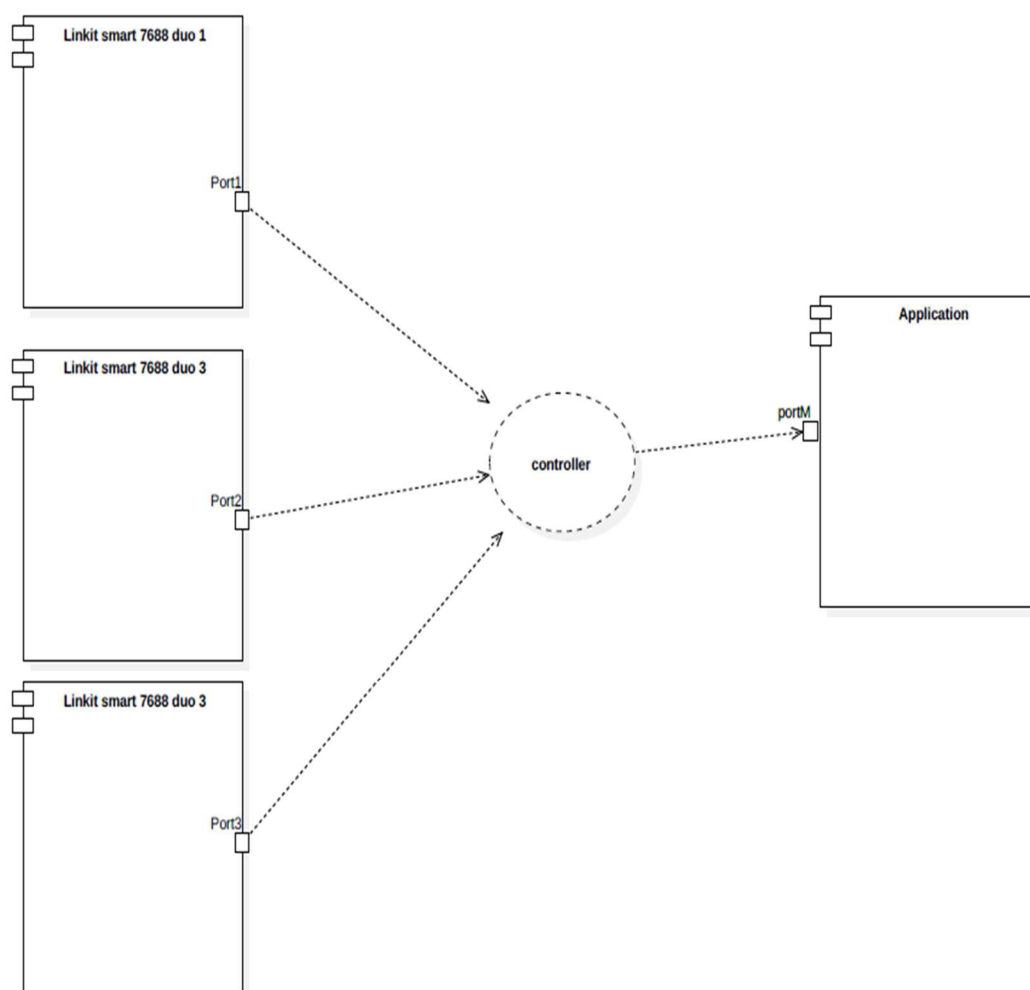
<http://www.jmlr.org/papers/v12/pedregosa11a.html>

ב. תרשימים וטבלאות

היות וזה נושא מחקרי, לא עומדים לרשותנו כרגע כל הידע והמידע שנצטרך להשלמת הפרויקט ולכן רוב התרשימים שביצענו הן ברמת ההשערה לגבי מה אנחנו מבינים שהמערכת תהיה, תתנהג ותיבנה על ידנו.

תרשים רכיבים – components

המערכת הכללית תורכב מרשת של חיישנים שתעביר את המידע למרכז הבקרה דרך צינור מידע אשר יעבור במתווך מסוים שיקשר ביניהם, וידע גם להעביר את המידע באופן אוטונומי לחיישנים האחרים.



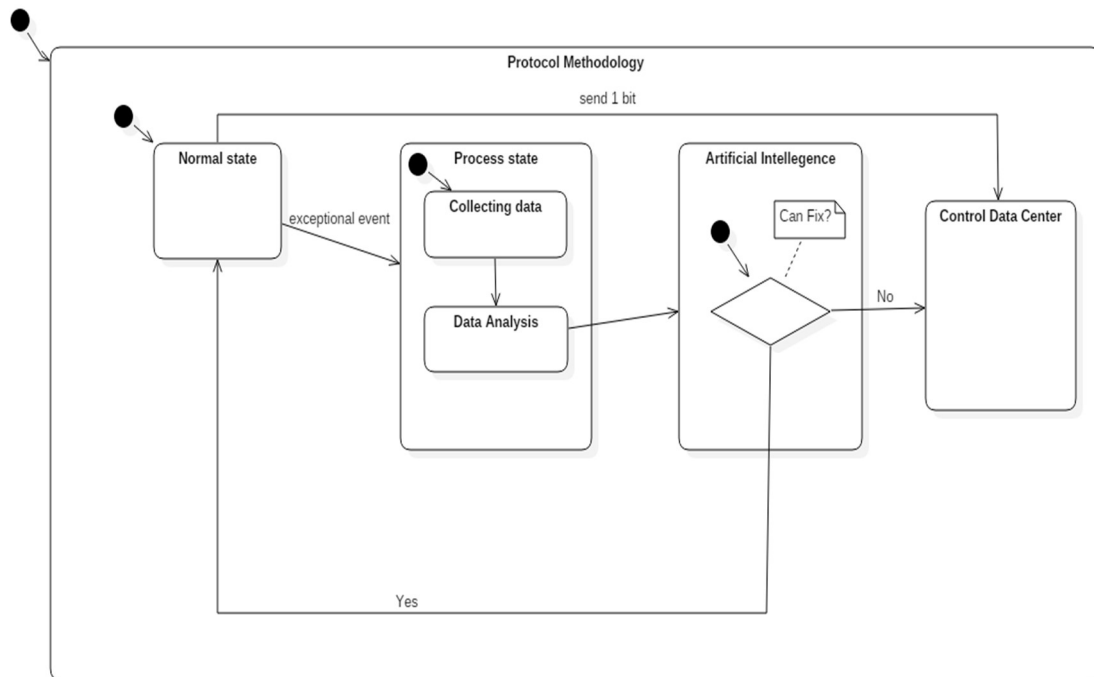
תרשים מצבים - State Chart

בתרשים הבא נתאר את המערכת במצבי הפעילות שלה.

המערכת בברירת מחדל תפעל במצב תקין, במצב זה כדי גם להעיד שהמצב תקין תשלח ביט בודד באופן תמידי או לסירוגין (פעם בכמה דקות למשל), ביט זה כמובן לא יכול מידע נוסף כי פשוט אין צורך.

במקרה של אירוע חריג אותו תזהה המערכת, המערכת תעבור למצב אירוע, תאסוף מידע אשר יגיע מהחיישן, המידע יאובחן ובשלב זה האלגוריתם יבדוק האם ביכולתו לתפעל את האירוע ולתקן את הבעיה או לבדוק האם האירוע לא חמור ולא מצריך מעבר של מידע למרכז הבקרה ולגורמים האנושיים.

אם כן המערכת חוזרת לשגרה וממשיכה בשליחת ביט בודד, אם אין ביכולת האלגוריתם לתקן את הבעיה המערכת שולחת למרכז הבקרה (גורמים אנושיים) חבילת מידע עם פירוט נרחב על האירוע שקרה.



ג. תכנון הפרויקט

| | |
|--|---|
| פגישת הכרות עם המנחה | 9.10 |
| בחירת נושא והגשה לרכז | 10.10 |
| הגשת הפרויקט וטופס ההתנעה | 22.10 |
| פגישת עבודה עם המנחה על שלב ההצעה | 30.10 |
| הגשת דו"ח שלב ההצעה: סקר שוק, דרישות, מהות הפרויקט, סיכונים ותרשימים. | 19.11 |
| הגשת מוצר אלפא – מימוש אב טיפוס, מסמכים נוספים וארכיטקטורת המערכת. | סוף סמסטר א' |
| הגשת מוצר ביטא – תיכון ומימוש האלגוריתם, גרסת ביטא: ניסוי של שינוי מצב | אמצע סמסטר ב' (תאריך שיקבע ע"י הרכז) |
| שיפור האלגוריתם לאחר הפקת לקחים ודגשים מהמנחה | שבועיים לאחר הגשת מוצר ביטא |
| הגשת הפרויקט | סוף סמסטר ב' |
| פגישת הפקת לקחים וסיכום הפרויקט | תאריך שיקבע עם המנחה |

כמו כן יקבעו פגישות רבות עם המנחה במהלך השנה על מנת לתכנן תכניות, משימות ועוד.

ד. טבלת סיכונים

| # | הסיכון | חומרה | מענה אפשרי |
|---|---|---------|---|
| 1 | לא נגיע לתוצאה שרצינו בסיום הפרויקט | גבוהה | ייעוץ נרחב עם המנחה, מעקב צמוד אחר ההתקדמות של הפרויקט בהתאם לציפיות מכל שלב, במקרה הגרוע תיאום ציפיות חדש. |
| 2 | סביבת העבודה וכלי פיתוח לא יהיו מענה ראוי למטרת הפרויקט | גבוהה | מחקר מעמיק על כלי הפיתוח, התאמת השבב לסביבת העבודה, במקרה הגרוע החלפת שבב. |
| 3 | חילוקי דעות עם המנחה לגבי מטרות ותוצאות רצויות | בינונית | עבודה משותפת ופגישות מרובות להכוונה ותיאום ציפיות עם המנחה. |

| | | | |
|---|---|---------|---|
| 4 | האלגוריתם יביא לתוצאות לא צפויות | בינונית | מתן זמן רב לתיקון שגיאות וריג'קטים , בדיקות מרובות במגוון מקרי קצה ועוד. |
| 5 | האלגוריתם לא יזהה אירוע חריג שיעיד על שינוי מצב | גבוהה | תיקון שגיאות , כתיבת האלגוריתם בצורה מודולרית ומסודרת שתאפשר חזרה לאחור מבלי לפגוע ביעילותו |
| 6 | מוגבלות זיכרון תפגע ביעילות האלגוריתם | בינונית | צמצום האלגוריתם , צמצום ספריות שפחות חשובות לצורכי המערכת , עבודה שוטפת על הקוד בדגש על יעילות מקום וזמן. |

ה. טבלת דרישות

| מס' דרישה | תיאור | סוג |
|-----------|---|-------------|
| 1 | סביבת העבודה תזהה את השבב ותצליח לכתוב את האלגוריתם על השבב | פונקציונאלי |
| 2 | השבב יצליח להפעיל את האלגוריתם ביעילות מקום מינימלית | פונקציונאלי |
| 3 | השבב ידע לעבוד במצב תקין ולזהות שינוי מצב ולהעביר מידע למרכז בקרה | פונקציונאלי |
| 4 | האלגוריתם יהיה מודולרי כך שניתן יהיה להוסיף אלמנטים נוספים בעתיד שישדרגו את האלגוריתם | תמיכה לעתיד |
| 5 | האלגוריתם יהיה יציב לבדיקות מרובות וימנע קריסות מערכת | יציבות |
| | | |