



פרוייקט גמר

לתואר הנדסאי במגמת _____ אלקטרוניקה

שם הפרוייקט: _____ פאנל סולרי מתכוונן

שם הסטודנטים: _____ דניאל לוין ואריאל פלקון

העבודה בוצע בהנחיית: _____ שמעון פיטלסון

מקום ביצוע העבודה: _____ המכללה הטכנולוגית אורט בראודה (כרמיאל)

תאריך הגשה: _____

הצהרת הסטודנט

אני	דניאל לוין	- ת.ז.	<u>315048587</u>
אני	אריאל פלקון	- ת.ז.	<u>318280368</u>

החתומים מטה, מצהירים בזאת, שכל עבודת הגמר המוגשת בחוברת זו הנה פרי עבודתנו בלבד, על בסיס הנחייתו של המנחה, ותוך הסתמכות על מקורות הידע והמידע האחרים המצוינים בביבליוגרפיה המובאת בסוף חוברת זו.

אנו מודעים לאחריות שאנו מקבלים על עצמינו ע"י חתימתנו על הצהרה זו שכל הנאמר בה הינו אמת ורק אמת.

חתימת מגיש העבודה

חתימת מגיש העבודה

אישור המנחה:

הריני מאשר הגשת החוברת להערכה

חתימת המנחה

אישור ראש המגמה:

הריני מאשר הגשת החוברת להערכה

חתימת ראש המגמה

נספח מס' 1 (11-4-51)

תאריך: 29.11.2019

הצעה לפרויקט גמר

שם המכללה : המכללה הטכנולוגית אורט בראודה, כרמיאל

מגמה : הנדסת אלקטרוניקה

מסלול : הנדסאים


(2695) 10

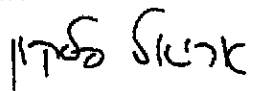
שם התלמיד	ת.ז. 9 ספרות	כתובת	טלפון	תאריך סיום הלימודים	ביצוע הפרויקט
דניאל לוי	315048587	שלום עליכם 24, עפולה	0507168509	04/20	אורט בראודה
אריאל פלקון	318280368	האלה 6/4, טירת הכרמל	0527410067	04/20	אורט בראודה

שם המנחה	כתובת	טלפון	תואר	מקום עבודה/תפקיד
שמעון פיטלסון	איה 46 כרמיאל	0524805333	B.Sc,	מכללת אורט בראודה מרצה/מנחה פרויקטים

שם הפרויקט: פאנל סולרי מתכוונן

- בקר אשר ישלח פקודות למערכת מנועים אשר יזיזו פאנל סולרי אל הכיוון בה השמש נמצאת.
- סדרת חיישנים אשר יזהו את מיקום השמש וישדרו את הנתונים הללו אל הבקר.
- ישודרו נתונים שונים לגבי המערכת אל הפלאפון של המשתמש בצורה אלחוטית.

תאריך: 18.10.2019 חתימה: 

תאריך: 18.10.2019 חתימה: 



חתימת המפקח הארצי

חתימת המנחה

חתימת הבוגר

נספח מס' 1 (11-4-51)

01-3-11 (08/06)

בית ליאון גולדברג, דרך מנחם בגין 86, תל אביב, ת.ד. 36049, מיקוד 67138, טלפון: 5634111-03

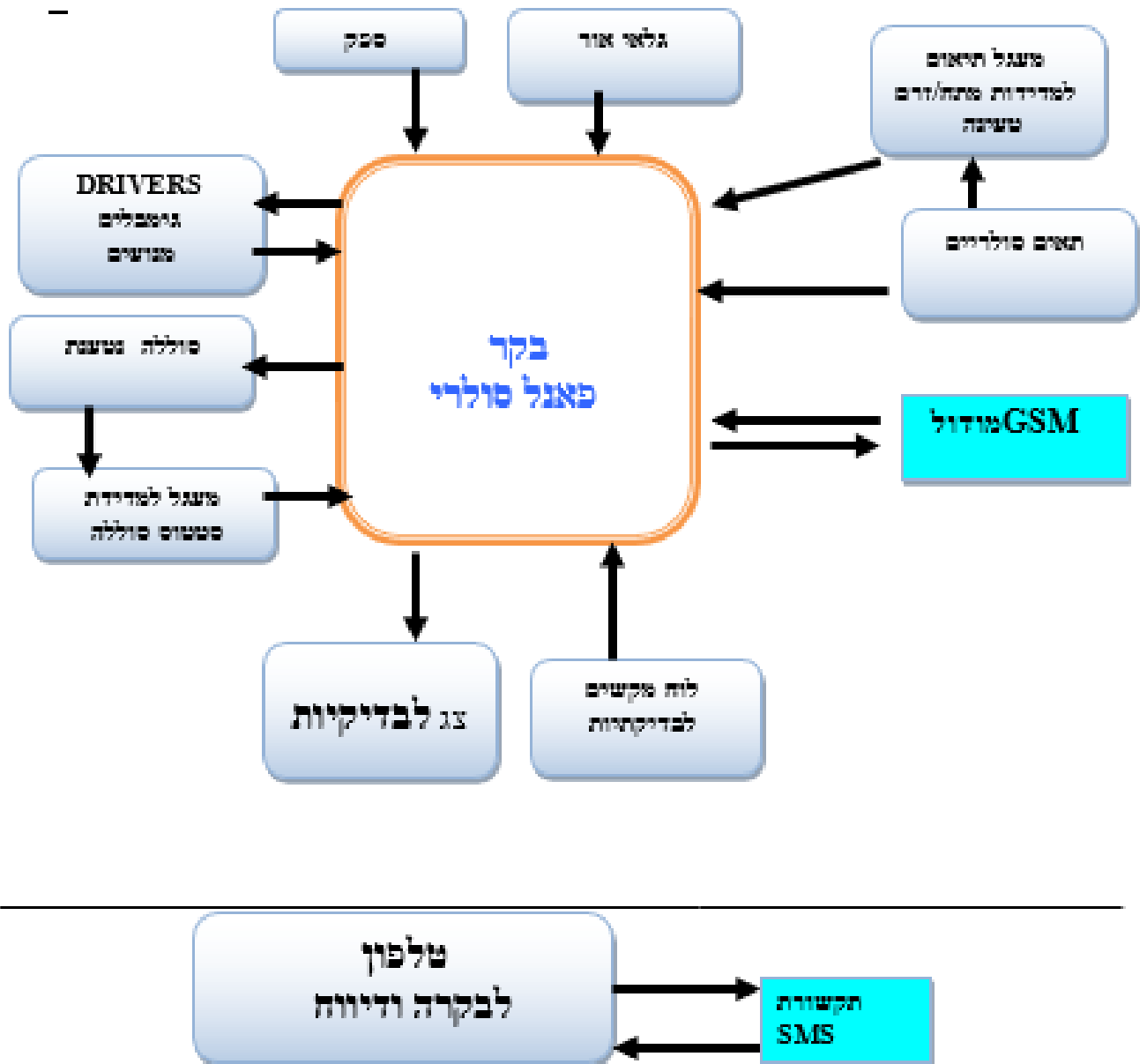
יחידת הפרויקטים - טלפון: 5634205-03 פקס: 5634251-03

תיאור קצר ונתונים טכניים (בצירוף סקיצה)

מפרט מכני:

1. בקר ממשפחת ARDUINO
2. מודול GSM לתקשורת
3. גימבלים, DRIVERS ומנועים
4. סוללה נטענת
5. תאים סולריים
6. מקשים וצג לבדיקות
7. מעגל תיאום למדידת מתח/זרם הטעינה
8. מעגל למדידת מצב סוללה
9. גלאי עצמת אור
10. ספק

סכמת מלבנים :



חלוקת אחריות

סטודנט דניאל לוי ת.ז. 315048587 הבקר, מנועים. גימבלים, בקרה

סטודנט אריאל פלקון ת.ז. 318280368 GSM, חיישנים. תאים סולריים וסוללה

שם הסטודנט דניאל לוי ת.ז. 315048587 חתימה: דניאל לוי

שם הסטודנט אריאל פלקון ת.ז. 318280368 חתימה: אריאל פלקון

בטלסון

חתימת המנחה

הערות מרכז המגמה במכללה

אישור מרכז המגמה:

שם:

הערות המפקח הארצי

דניאל מור

ראש המחלקה לאלקטרוניקה
המכללה הטכנולוגית להנדסאים
אזור ברענה כרמיאל תאריך:

28/4/19

אישור המפקח הארצי:

שם:

לשכת מפקח

חתימה:

א

תאריך:

28/4/19

הבעת תודה

אנו רוצים להביע הערכה לכלל המרצים אשר לימדו אותנו,
תגברו אותנו ועזרו לנו להצליח להגיע עד הלום.

תודה מיוחדת למר דויד מור אשר תמך בנו ועודד אותנו לשאוף
למצוינות והישגים גבוהים, ועזר לאורך כל הדרך.

כמו כן, אנו רוצים להודות למנחה הפרויקט שלנו שמעון פיטלסון אשר
עזר לנו רבות בפרויקט, והנחה אותנו בצורה נאמנה ומקצועית ביותר.

בנוסף על כך, אנו מביאים הערכה כלפי חברינו לכיתה אשר נחלצו
לעזרתנו בכל בעיה ובקשה ברכיבים או בפרטים טכניים.

וכמובן תודה ניכרת לחברת רפאל אשר נתנה לנו ההזדמנות להיות
חלק מהפרויקט הייחודי הזה, ובפרט לדנה מרפאל אשר עזרה לנו
בכל אשר היינו צריכים בתור כיתה וליוותה אותנו לאורך כל הדרך.

תקציר

הפרויקט הינו מערכת סולארית, אשר מתכוונת באופן אוטומטי כלפי מיקום השמש (מקור האור).

המערכת מייעלת את כושר קליטת קרני השמש במהלך היום/עונה/שנה.

המערכת הסולארית מתוכננת לייעל את כושר קליטת קרני השמש ע"י סיבוב והזזה של הפאנלים הסולאריים על גבי שני צירים (ציר אופקי וציר אנכי), בעקבות קליטה של מערך החיישנים ובהתאמה.

המערכת תציג את הנתונים על גבי ממשק המשתמש בשני אופנים:

1. המערכת תציג נתוני מתח, זרם, הספק ואחוז טעינה של הסוללה הנטענת מהמערכת הסולארית על גבי צג המשתמש.
2. המערכת תשלח נתונים (לעיל) לפלאפון המשתמש כהודעת SMS.

המערכת תחובר באופן ישיר למטרת שימוש האנרגיה אשר הפאנל מייצר, לדוגמא: חיבור לפלאפון.

תוכן העניינים

<u>העמוד</u>	<u>הנושא</u>
1	תוכן העניינים
2	רשימת טבלאות
3	רשימת שרטוטים
4	<u>פרק 1 - מבוא</u>
5	1.1 תיאור הפרוייקט
6	1.2 סקירה מקצועית
7	1.3 הוראות הפעלה
8	1.4 מפרט טכני
9	<u>פרק 2 – מבנה עקרוני</u>
10	2.1 תרשים מלבנים עקרוני
12	2.2 תרשים זרימה עקרוני
14	<u>פרק 3 - חומרה</u>
15	3.1 תרשים חשמלי מפורט
17	3.2 פירוט רכיבים
34	<u>פרק 4 - תכנה</u>
35	4.1 תרשים זרימה ואלגוריתמים של תתי שגרות
41	4.2 קובץ LIST של התוכנה
45	<u>פרק 5 – סיכום ומסקנות</u>
46	5.1 מדידות ותוצאות
50	5.2 מסקנות
51	<u>פרק 6 - ביבליוגרפיה</u>
52	<u>פרק 7 - נספחים</u>
53	7.1 רשימת רכיבים
54	7.2 נתוני יצרן

רשימת טבלאות

העמוד

6

54

הנושא

סקירה מקצועית

רשימת רכיבים

רשימת שרטוטים

<u>העמוד</u>	<u>הנושא</u>
10	תרשים מלבנים עקרוני
12	תרשים זרימה עקרוני
15	שרטוט חשמלי מלא
17	מעגל לזיהוי מקור אור
18	מעגל להפעלת מנועי סרבו
19	מעגל להפעלת מסך
20	מעגל למערך הסולרי
21	מעגל לתקשורת
35	תרשים זרימה כלל מערכתי
36	תת שגרה מדידת אחוז סוללה
37	תת שגרה תצוגת מסך
38	תת שגרה רכיב GSM
39	תת שגרה זיהוי מקור אור ותזוזה אליו
40	תת שגרה מדידת מתח/זרם

פרק 1 – מבוא

1.1 תיאור הפרויקט:

הפרויקט הינו מערכת סולארית, אשר מתכווננת באופן אוטונומי כלפי מיקום מקור האור, על מנת לקבל תפוקת אור מרבית, ועל ידי כך תספק הזנה חשמלית בצורה היעילה ביותר.

המערכת כוללת בתוכה סדרת חיישנים ותת מערכות, אשר מאפשרות לה לפעול ומספקים למשתמש מידע כגון: אחוז טעינת הסוללה, מתח חשמלי ישיר (DC), זרם ואת ההספק אשר מספקת המערכת הסולארית לסוללה.

כמו כן, המערכת מכילה בתוכה תקשורת סלולרית (GSM), אשר תאפשר למשתמש לשם קבלת נתונים אודות תפקוד המערכת בהיבטים שונים, כגון: אחוז טעינת הסוללה, מתח טעינת הסוללה ונתונים שונים אחרים, על ידי שליחת הודעת מערכת למכשיר סלולרי.

מערכת זו מיועדת לסייע לחברות ואנשים פרטיים לקבל את המיטב מן המערכת הסולרית ביחס למערכות קיימות בשוק, ולקדם ליותר אנשים לעבור לשימוש ירוק בין היתר במערכות סולריות, על מנת לשמור על הסביבה על ידי שימוש יעיל באנרגיה מתחדשת.

1.2 סקירה מקצועית

קיימות כמה וכמה מערכות כאלו בשוק כיום, לרוב אלו חברות מסחריות אשר מספקות את המוצר לרוב, לגופים עסקיים ולגורמים פרטיים בתנאי החברה המספקת.

בחלק זה אנו נציג מספר מערכות דומות, אשר יהיו שונות בצורתן החיצונית ובמערכת ההפעלה שלהן מהמערכת הסולארית שלנו, אך עקרון הפעולה שלהם כמעט וזהה למערכת שלנו.

קריטריונים	מונה נטר	מערכת סולארית עצמאית	מערכת סולארית ביתית	דגם הפרויקט
הספק	20 עד 30 ואט בממוצע לכל מ"ר.	20 עד 30 ואט בממוצע לכל מ"ר.	מערכת סולארית ביתית יכולה להגיע למקסימום של 15 קילוואט.	1W
טווח שטח	בד"כ מעל 100 מ"ר.	בד"כ מעל 100 מ"ר.	כגודל הגג.	כגודל המערכת.
מיקום התקנה	ניתן לבנות מערכת סולארית על גג הבית/ בית ספר/ מפעל/ בניין.	מערכת שבמרבית המקרים ממוקמת על גבי גגות באזורים שאינם מחוברים לרשת החשמל.	ניתן להתקין את המערכת על גבי כל סוגי הגגות, בטון, רעפים וגגות מתכת.	מערכת ניידת.
עלויות	כ- 6,000 ש"ח לקילו וואט (6 ש"ח לואט).	*משתנה בהתאם לגודל המערכת, מיקום ההתקנה ואיכות חומרי המערכת.	נע בין 35,000 ש"ח ל- 120,000 ש"ח בממוצע.	כ- 250 ש"ח.
חסרונות המערכת	השקעה ראשונית גבוהה המתבטאת בעלות גבוהה לקילוואט בהשוואה לתחנות כוח קונבנציונליות.	תפיסת שטח רב יחסית להפקת חשמל בשיטה הקונבנציונליות בפחם, דלק נוזלי או גז.	הפקת אנרגיה לא סדירה בימים מעוננים ובליילה בתחנות כוח ללא אגירה או מקור אנרגיה נוסף.	מערכת כזו בפועל תעלה יותר משום שזו מערכת דו צירית מבוקרת בעלת משוב.
יתרונות המערכת	מאפשרת לכל צרכן חשמל להקים לעצמו מערכת סולארית אשר תייצר חשמל לשימוש עצמי ללא הגבלה על משך ההתקשרות וכמעט ללא הגבלה על גודל המערכת.	מדובר באנרגיה ירוקה והיעילה לסביבה, על כן איננה מזדהמת את הסביבה כמו מערכת חשמל רגילה. כמו כן, המערכת איננה תלויה בדלק או במשאבי טבע יקרים אחרים שנוכחותם בעולם לא תמיד תהיה שכן נעלמים והולכים, אלא מתבססת על משאב טבעי שקיים באופן תמידי ללא עלות- השמש.	ייצור חשמל אשר אינו מזדהם את הסביבה – הייצור אינו פולט גזים רעילים, אינו יוצר רעש, אינו גורם להתחממות של כדור הארץ ואינו מכלה משאבים – שכן השמש היא משאב בלתי מוגבל. היתרון הכלכלי – אנרגיה שמש ניתנת בחינם, מה שחוסך את התשלומים לחברת החשמל במרבית שעות היום.	מייצלת את "קליטת" קרני השמש במהלך היום/שנה באופן אופטימלי ואינה קבועה לזווית אחת.
רווח-קיץ	בקיץ, עבור הזרמה במתח נמוך ניתן לקבל 1.25 ש"ח לקוט"ש.	אי תשלום לחברת החשמל.	1.9 שקלים לקילו- וואט (KW).	-----
רווח-תקופת מעבר	בתקופת מעבר 52- אגורות לקוט"ש.	אי תשלום לחברת החשמל.	1.67 שקלים לקילו- וואט (KW).	-----
רווח-חורף	בחורף כ- 38 אגורות לקוט"ש.	אי תשלום לחברת החשמל.	1.65 שקלים לקילו- וואט (KW).	-----

1.3 הוראות הפעלה

הפעלת המערכת תיעשה על ידי חיבור המערכת למקור חשמל.

לאחר חיבור המערכת למקור הכוח המערכת תפעל בצורה אוטומטית ותמקם את עצמה אל מקור האור.

בכדי לטעון את מכשיר הסלולר נדרש לחבר אותו אל חיבור ה-USB.

על מנת לבקש שליחת נתונים מן המערכת אל מכשיר הסלולר נדרש לשלוח הודעת SMS אל המספר: 052-741-0067 אשר מכילה את ההודעה: "P".

ניתן לראות את אחוז טעינת הסוללה על ידי לחיצה על הכפתור ליד מסך ה-LCD ומערך של 4 לדים יספקו אינדיקציה לטעינת הסוללה.

כמו כן אם נדרש, בעת לחיצה ארוכה על הכפתור, ידלק פנס ולאחר לחיצה ארוכה נוספת הפנס יכבה.

1.4 מפרט טכני

מיקרו בקר Arduino Mega 2560:

- משתמש בבקר ATmega2560.
- 54 פני I/O (דיגיטליים).
- 16 פנים אנלוגיים.
- 15 ערוצי PWM.
- זרם מקסימלי לפין דיגיטלי הוא 40mA.
- מתח עבודה של 5V.
- 256KB של זיכרון.
- 8KB זיכרון SRAM.
- 4KB זיכרון EEPROM.
- מהירות שעון של 16MHz.
- מתח הזנה מומלץ בין 7-12V.
- גבול מתח הזנה בין 6-20V.

פאנלים סולריים OVERFLY:

- הספק של 1W.
- מתח תפוקה מקסימלי של 6V.

מנועי סרבו מסוג SG90:

- מומנט מרבי של 1.8kgF·cm.
- מהירות עבודה של 60 מעלות כל 0.1 שניות.
- מתח עבודה בין 4.8 וולט ל- 6 וולט.
- טמפרטורת עבודה בין 0 ל-50 מעלות צלזיוס.

רכיב SIM800L GSM:

- מתח הזנה נע בין 3.4 ל- 4.4 וולט.
- טמפרטורת עבודה נעה בין (40° -) ל- 85° מעלות צלזיוס.

רכיב למדידת מתח וזרם INA219:

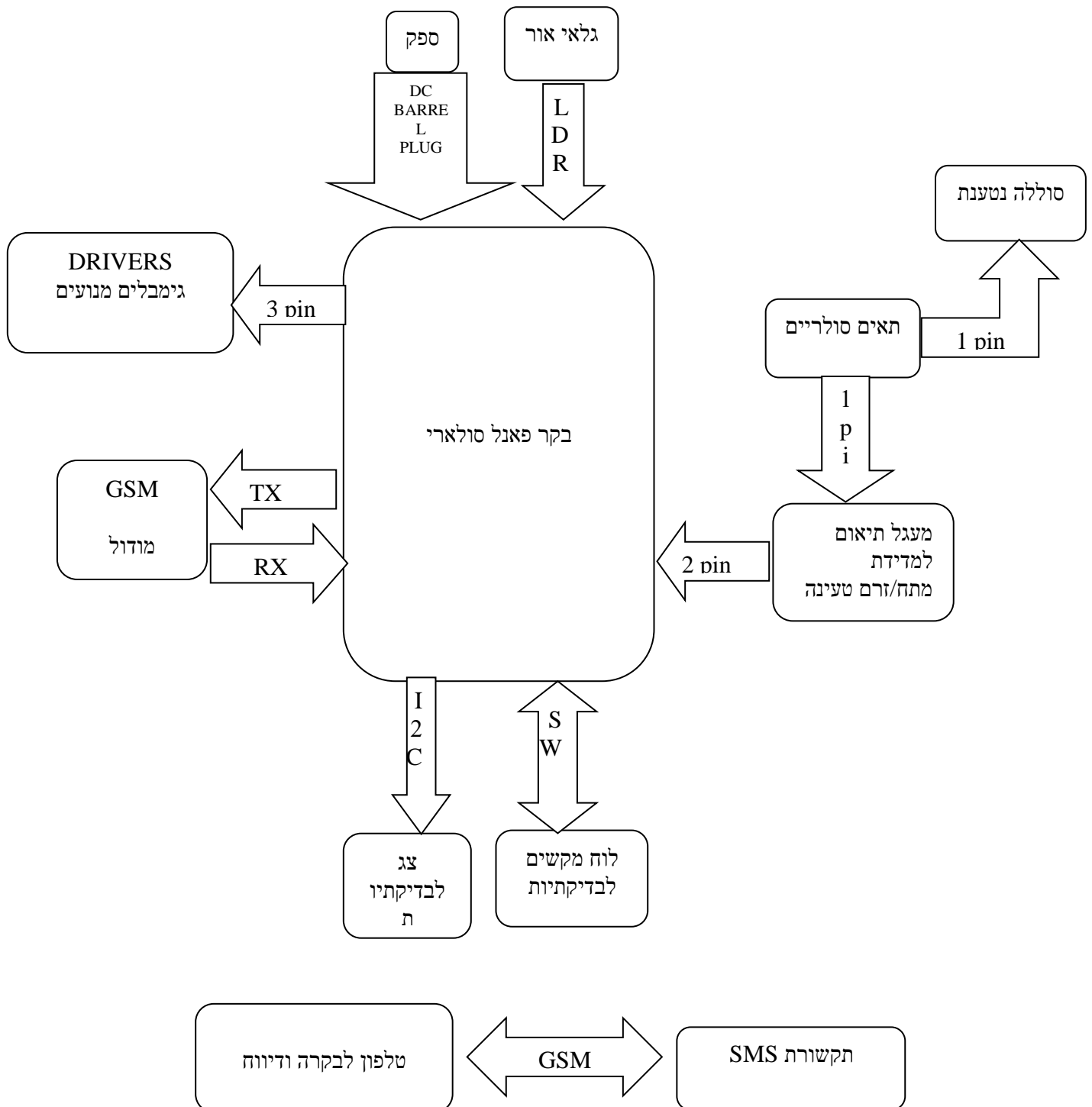
- מתח עבודה נע בין 3 ל-5.5 וולט.
- טמפרטורת עבודה נעה בין (40° -) ל- 125° מעלות צלזיוס.
- טווח מדידת מתח ±26V.
- טווח מדידת זרם ±3.2mA.

מסכי LCD 1602 I2C:

- מתח הזנה של 5V.
- טמפרטורת עבודה בין (20-) ל- 70 מעלות צלזיוס.

פרק 2 – מבנה עקרוני

2.1 תרשים מלבנים עקרוני



הסבר התרשים:

בקר פאנל סולרי Arduino:

זהו בעצם הבקר שלנו אשר מפעיל את כול המערכת, תפקידו העיקרי הינו לנהל את תפעולי המערכת כמו: איסוף מידע מן החיישנים, הזזת הלוחות הסולריים אל מקור האור, הצגת נתונים על המסך ותקשורת עם המשתמש, הבקר יכול את התוכנה האחראית לפעילויות השוטפות של המערכת.

גלאי אור LDR:

גלאי האור משנים את התנגדותם בהתאם לכמות האור אשר הם קולטים בעזרתם אנו בעצם מקבלים קריאה על מיקום השמש (מקור האור) בהשוואה למיקומם.

ספק:

תפקידו של הספק הוא לספק את מתח ההזנה לבקר ארדואינו שלנו. מנועי סרבו:

מנועי הסרבו הם אלו שמניעים לנו את הפאנל הסולרי ומשנים לו את המיקום על מנת שיפנה אל מקור האור.

GSM:

זהו הרכיב תקשורת שלנו אשר מאפשר לנו לשלוח ולקבל הודעות SMS, מן המשתמש למערכת, שכוללים בתוכם נתונים על התפוקה החשמלית של הפאנל הסולרי, מתח, זרם והספק.

מסך LCD:

תפקידו של המסך הוא להציג לנו נתונים מספריים לגבי המערכת וכמו כן את אחוז טעינת הסוללה.

תאים סולריים:

אלו הם הפאנלים הסולריים אשר נטענים ממקור האור ומספקים לנו טעינה לעומס (סוללה נטענת).

מעגל מדידת זרם מתח INA219:

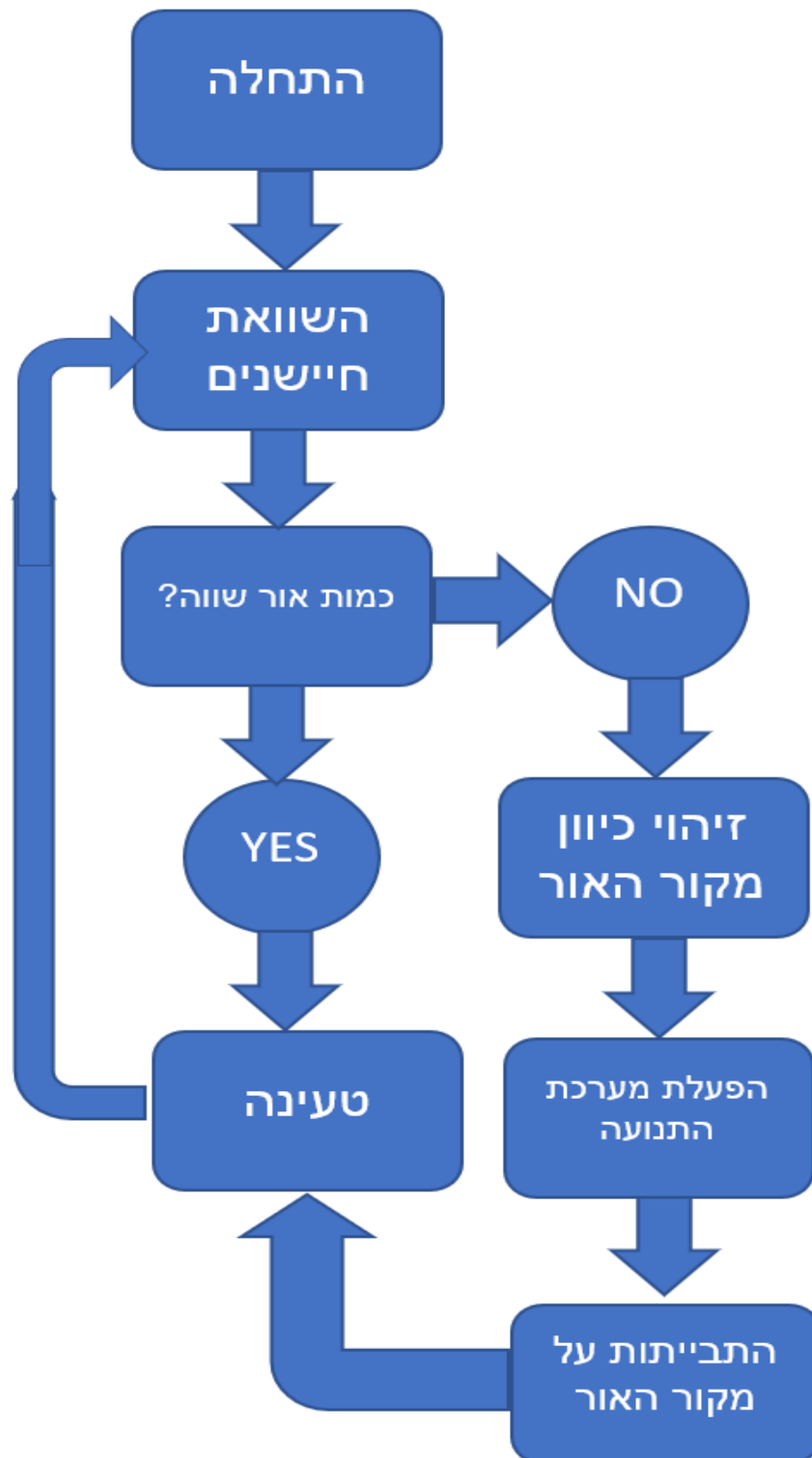
תפקידו של רכיב זה הוא למדוד ולחשב לנו את הזרם והמתח אשר מגיעים מהלוחות הסולריים ולהציג לנו את הקריאה הזו לפעולות המשך.

סוללה נטענת:

העומס אשר אנו רוצים להטעין בעזרת הפאנלים הסולריים.

2.2 תרשים זרימה עקרוני:

התרשים מתאר את סדר הפעולות בצד המפעיל, החל מהדלקת המערכת וכלה בסיום פעולתו וסגירתו.



אסקור בקצרה את תרשימים הזרימה:

התרשימים מראה את הפעולה העקרונית של הפעולה העיקרית של התוכנית, זו שמופעלת כל עוד המערכת נמצאת בפעולה. התרשימים מתאר מבט על, על פעולת המערכת כל זמן שהיא דלוקה/ בפעולה.

כל הפעולות המתוארות בתרשימים הזרימה מבוצעות בהתאם להנחיות המוגדרות להן, אותן אתאר בהמשך הספר.

*הפעלת המערכת- פעולה זו נעשית באופן חד פעמי בתחילת פעולת המערכת וכל הכלול בה.

הפעלת החיישנים הראשונית נעשית באופן חד פעמי כהגדרתם ומתן זמן הפעלה מתאים וכדומה על פי דרישות על רכיב.

*בקרת תנועה, באופן כללי המערכת 'תמתין' לזיהוי והתביינות על מקור האור.

במהלך ה"המתנה" הזו הוא ממשיך כרגיל; סריקה, תנועה וכדומה, ברגע שיקבל אחת מן הדרישות, יפעל בהתאם להן.

המערכת מצוידת בסט חיישנים לצורך מאסף מידע מהשטח ובקרת תנועה.

מאסף המידע מן החיישנים נעשה בצורה מחזורית כאשר כל זמן מוקצב, תעדכן המערכת את המידע הקיים בו מכל חיישן.

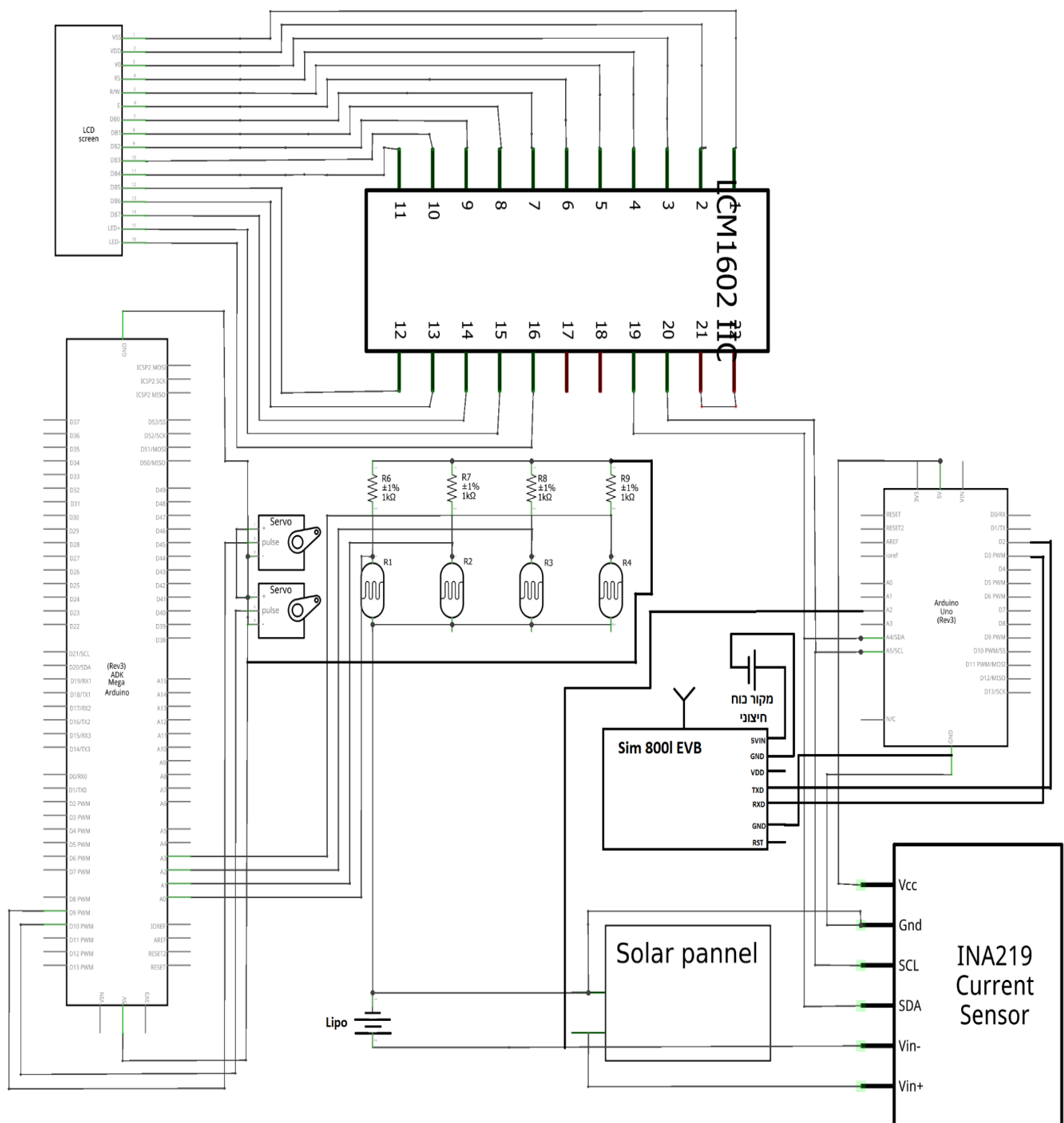
התרשימים מתאר את פעולת המערכת הכללית.

*המערכת תטעין את הסוללה אשר מחוברת אליה לצורך שימוש על ידי המשתמש.

*סיום- סיום הפעולה ייעשה בעת כיבוי המערכת על ידי ניתוק מהספקת החשמל.

פרק 3 – החומרה

3.1 שרטוט חשמלי מלא



3.1 הסבר לשרטוט החשמלי ולמעגלים

המערכת כוללת בתוכה 2 מיקרו-בקרים מסוג ארדואינו מגה וארדואינו אנו.
תפקיד הארדואינו אנו, הינו שליטה ובקרה עבור רוב רכיבי המערכת:
תצוגת מסך ה-LCD, רכיב התקשורת - Sim800l evb, רכיב מדידת מתח\זרם וקריאה מן הסוללה.

תפקיד הארדואינו מגה, הינו שליטה במערך התזוזה של המערכת, כלומר מחוברים אליו מנועי הסרבו אשר נותנים לנו את האפשרות להזיז את הלוחות הסולאריים ואת מערך ה-LDR, המוצמדים אל הפאנל הסולארי, אשר מזהים את מיקום מקור האור ובעזרת תהליך עיבוד המידע במיקרו-בקר, המיקרו-בקר מפענח את הימצאות מקור האור ביחס לנגדי האור.

3.2 רכיבים מיוחדים

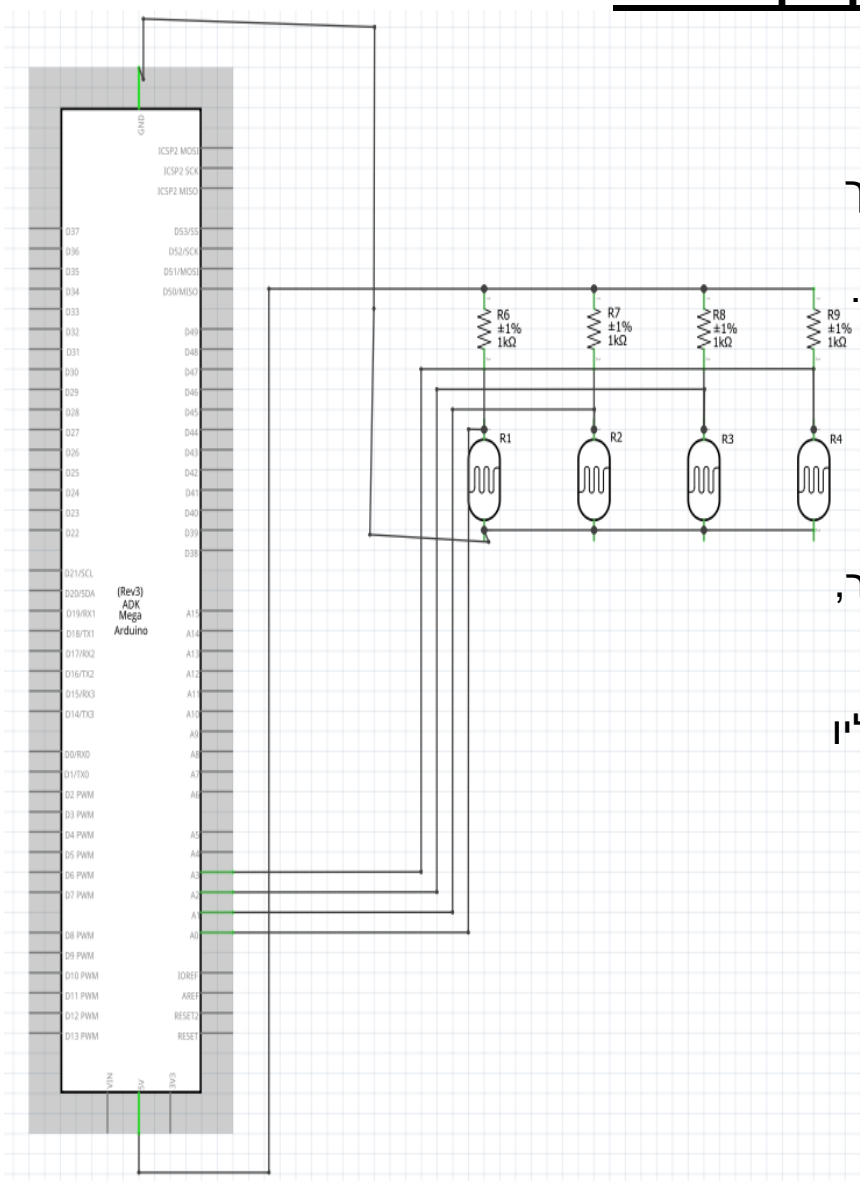
3.2.1 מעגל לזיהוי כיוון מקור האור:

המעגל הבא פועל על פי עקרון מחלק המתח, בין נגדים קבועים בעלי ערך של $1K\Omega$ ובין הפוטורזיסטורים (נגדי אור).

מיקרו-הבקר לוקח מנגדי האור קריאה אל הפורטים האנלוגיים. ככול שעוצמת ההארה על נגד האור תהיה נמוכה יותר, כך ההתנגדות שלו תהיה גבוהה יותר, ולפיכך המתח שיתפתח עליו יהיה גבוה יותר. ולהפך, בהתאמה.

הבקר ייקח את כל הקריאות מהפורטים וישווה בין התוצאות.

על פי השוואה זו נדע את מיקום מקור האור ביחס לנגדים וכמו כן ביחס לפאנלים הסולריים.

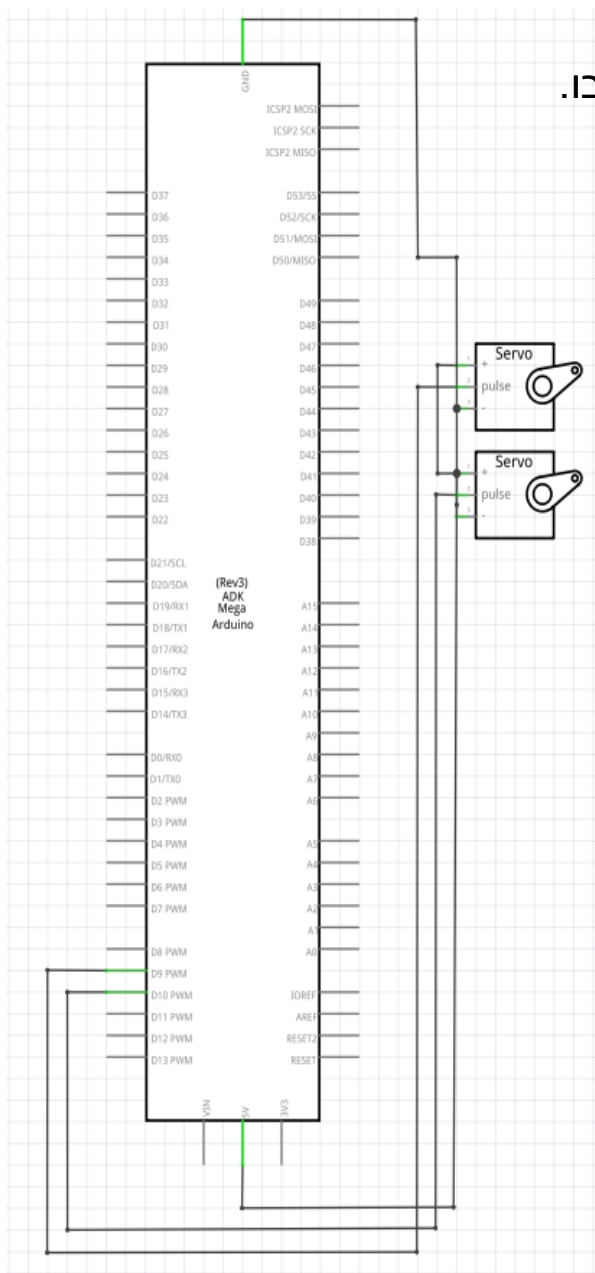


3.2.2 מעגל להפעלת מנועי הסרבו:

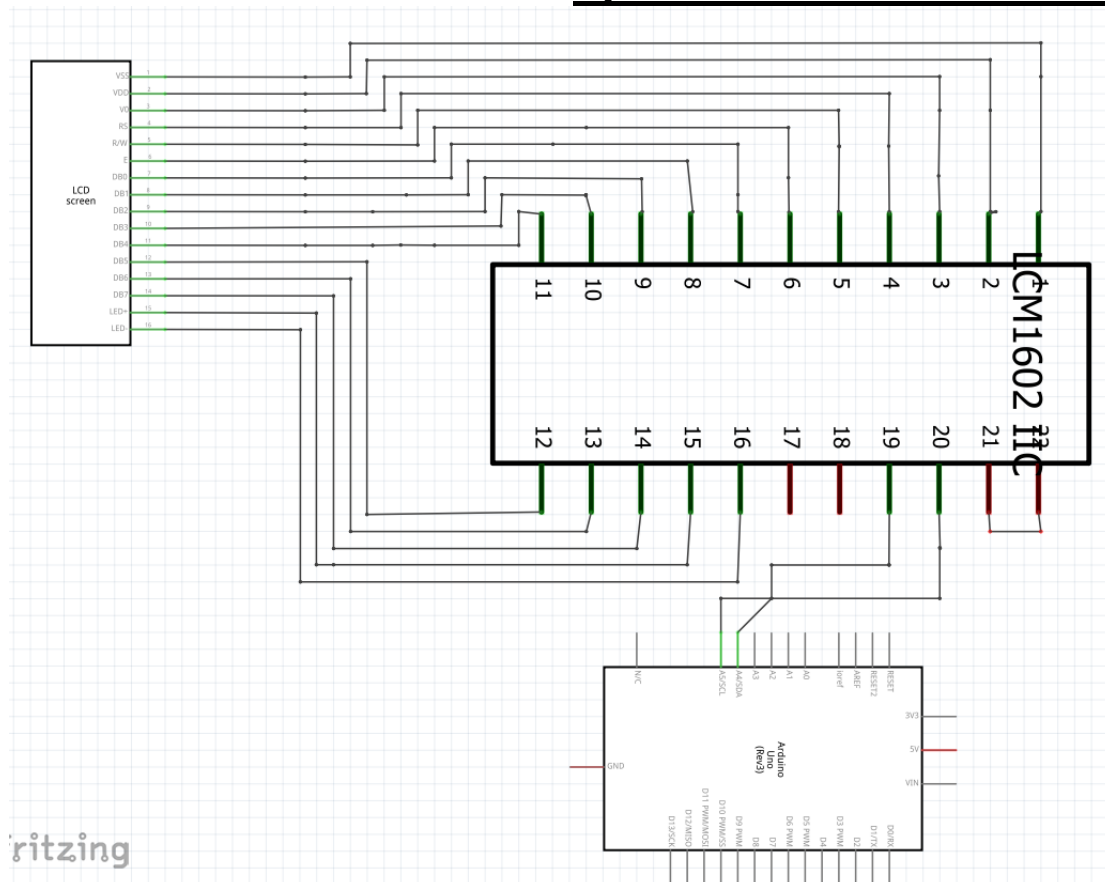
מעגל זה אחראי על פעולתם של מנועי הסרבו.
לכול מנוע סרבו קיימת כניסת PWM,
אשר חוברה לכניסה דיגיטלית של
המיקרו-בקר.

הכניסה הדיגיטלית משנה את ערכה
לפי הצורך הנדרש:

כאשר אנו נרצה תנועה בכיוון אחד,
אנו נקבל מצב יציאה בצורה מסוימת
וכאשר נרצה לקבל אותו לכיוון השני נקבל
את היציאה בצורה שונה.



3.2.3 מעגל להפעלת המסך:

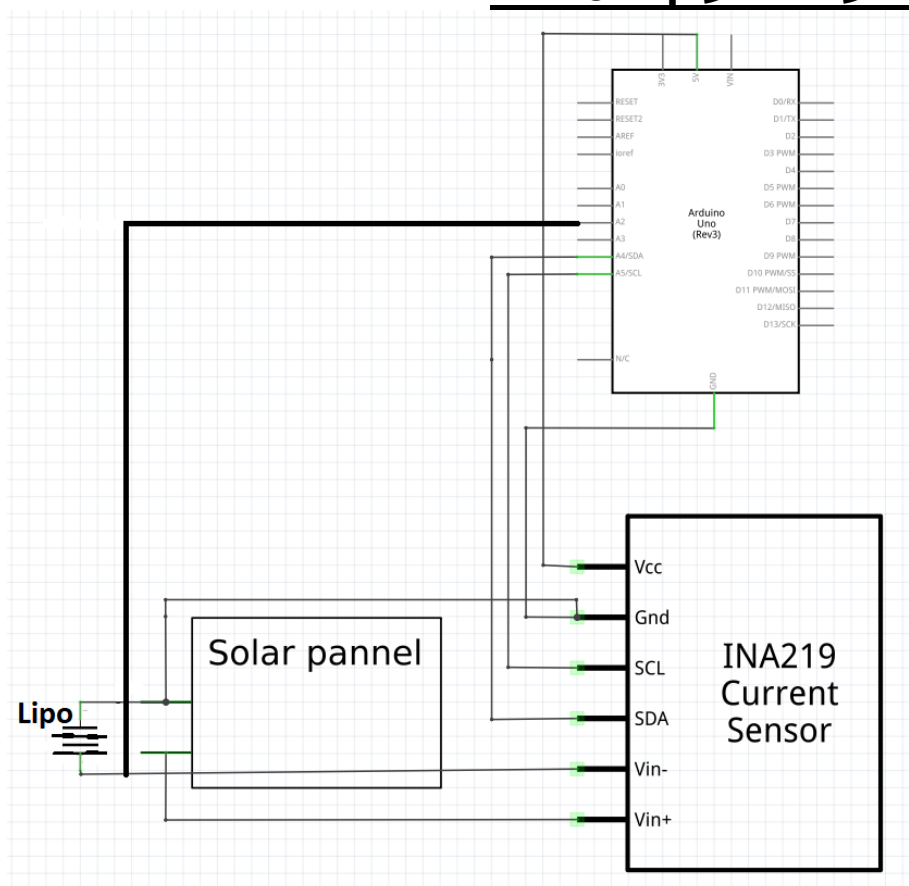


ritzing

המעגל להפעלת מסך ה-LCD, מפעיל לנו את מסך ה-LCD באמצעות רכיב ה-I2C. כניסות מסך ה-LCD מחוברות לרכיב ה-I2C, דבר שמייצל את השימוש במסך ה-LCD ומאפשר לנו לעבור ממצב שבו במקום להשתמש ב 16 חיבורים מהמסך אל רכיב המיקרו-בקר, למצב שבו מסך ה-LCD ימומש על ידי 4 חיבורים בלבד: מתח, אדמה, SCL ו- SDA.

תפקיד חיבורי ה- SDA וה- SCL, הינו לביצוע פעולות תקשורת עם רכיב הבקרה. תקשורת זו מתבצעת דרך שני קווי מידע, אשר מעבירים את המידע אל רכיב ה-I2C, והוא בתורו מתרגם את המידע המועבר אליו דרך קווי המידע, למידע אשר מוצג על מסך ה-LCD.

3.2.4 מעגל למערך הסולרי:

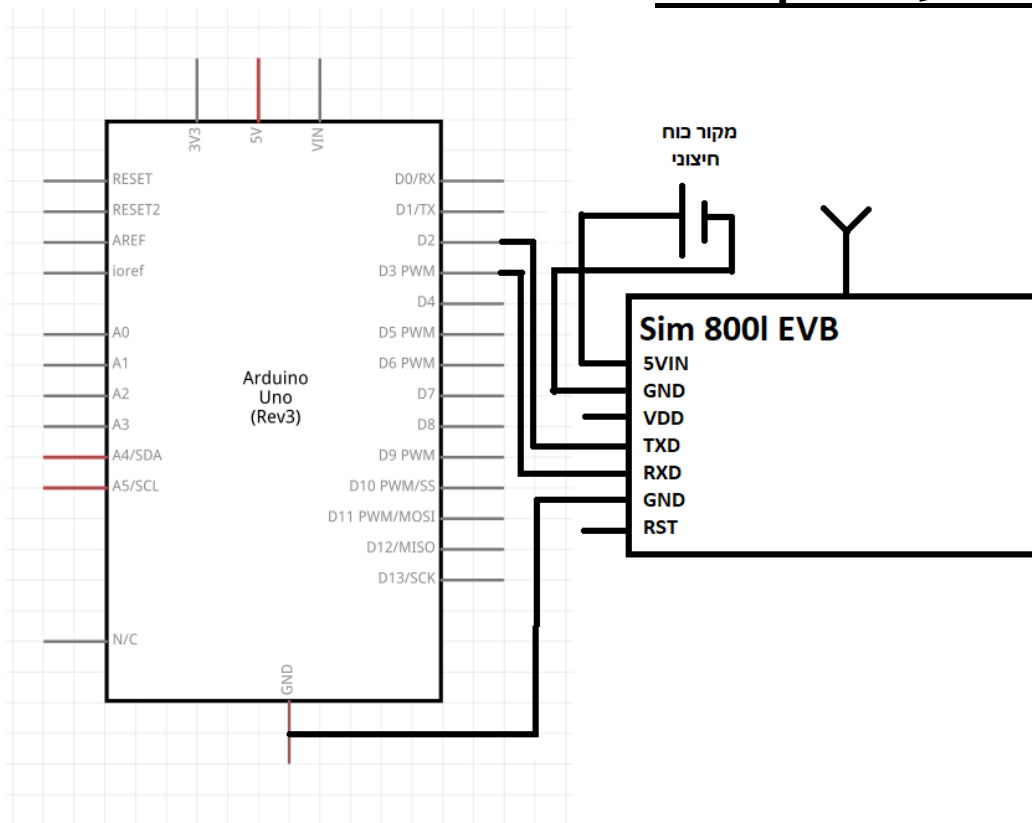


המעגל למערך הסולארי כולל בתוכו מספר רכיבים:
פאנל סולארי, מעגל למדידת מתח/זרם (INA219) וסוללת Lipo.
בחרנו לתחום את הרכיבים הנ"ל מכיוון ששישום מעגל נפרד לכול רכיב אינו פרקטי, משום ששלושת הרכיבים פועלים יחדיו.
הפאנל הסולרי מספק מתח וזרם, אשר עוברים דרך רכיב המדידה.
לאחר מכן רכיב המדידה מספק מידע אל המיקרו-בקר דרך חיבורי ה-SDA וה-SCL, דבר זה מתאפשר על ידי רכיב I2C המובנה בתוך הרכיב, מידע זה כולל בתוכו את ערכי המדידה של המתח.
את חישוב הזרם אשר עובר ברכיב נבצע בתוכנה, משום שאנו יודעים את גודל העומס החשמלי אשר עליו נמדד המתח.

באותו זמן הפאנל הסולרי מטעין לנו את העומס החשמלי, שבמקרה שלנו העומס הינו סוללת Lipo.

על מנת לדעת את אחוז טעינת הסוללה, המיקרו-בקר יקח קריאת מתח מן הסוללה, ודרך חישוב מתמטי בתוכנה אנו נדע את אחוז טעינת הסוללה.

3.2.5 מעגל לתקשורת:



תפקידו של המעגל המוצג לעיל הינו, להפעיל לנו את מערך התקשורת עם רכיב ה-GSM. זהו מעגל שבחיבורו של רכיב ה-SIM800I עם המיקרו-בקר, תפקידו לקבל הודעת SMS ולהשיב הודעה לשולח באמצעות מערך זה.

החלטנו לספק לרכיב GSM מקור מתח חיצוני משום שהארדואינו אינו מסוגל לספק לרכיב זרם גבוה מספיק. לרכיב GSM בו אנו משתמשים, נדרש מקור שמספק 2A אך הזרם המקסימלי שהארדואינו יכול לספק הינו 200mA.

חיברנו את רגל ה-TX (שידור Transmit) אל רגל דיגיטלית 2 של הארדואינו ואת רגל ה-RX (קליטה Receive) אל רגל דיגיטלית 3 של הארדואינו. חיבורים אלה מספקים לנו את התקשורת של הרכיב GSM עם הארדואינו ומאפשרים לנו שליחת או קבלת מידע מן כרטיס הסיים המחובר ברכיב.

3.3 הסברים על הרכיבים בשימוש:

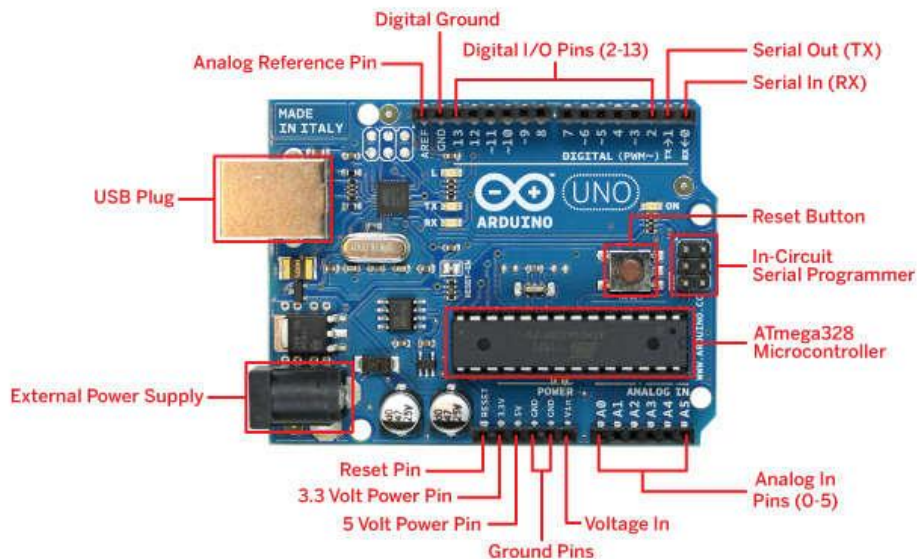
3.3.1 מיקרו-בקר ארדואינו:

מיקרו-בקר הארדואינו הינם למעשה ה"מוח" של המערכת כולה. רכיבים אלה מאפשרים למתכנן, לבצע שלל רחב של פעולות. פעולות אלו מתאפשרות על ידי כתיבת קוד לבקר הארדואינו דרך התוכנה המותקנת על המחשב, קוד זה מבוסס על שפת C.

בפרויקט הנ"ל אנו נשתמש ב- 2 מיקרו-בקרים, ארדואינו מגה ו- ארדואינו אונו.

להלן הסברים על כול מיקרו-בקר בנפרד:

מיקרו-בקר ארדואינו UNO:

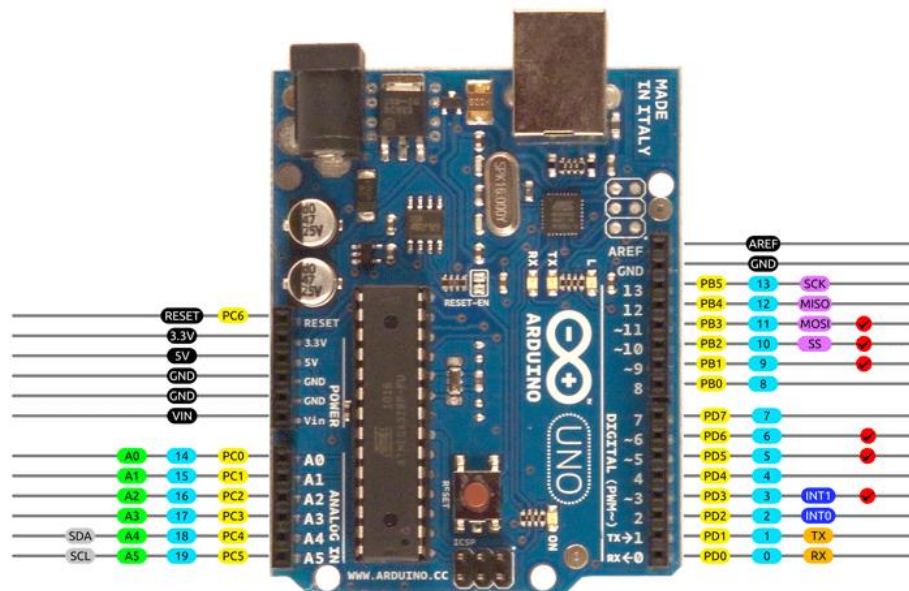


בקר הארדואינו UNO אשר באמצעותו ימומשו החלקים ה"עיקריים" בפרויקט, בקר זה שולט במספר תתי מערכות כגון: תקשורת, מדידה וכו'.

החלטנו לבחור בארדואינו UNO משום שהוא הארדואינו הנפוץ אשר החברה מייצרת, הוא קטן מימדים ומספק את רוב הפורטים הנחוצים למערכת.

מפרט טכני:

- משתמש בפקר ATmega328P.
- 14 פני I/O (דיגיטליים).
- 6 פנים אנלוגיים.
- 6 ערוצי PWM.
- הזרם המקסימלי לפין: 40mA.
- מתח עבודה: 5V.
- 32KB של זכרון.
- 2KB זכרון SRAM.
- 1KB זכרון EEPROM.
- מהירות שעון של 16MHz.
- מתח הזנה מומלץ בין 7-12V.



AVR DIGITAL ANALOG POWER SERIAL SPI I2C PWM INTERRUPT

Arduino Uno R3 Front

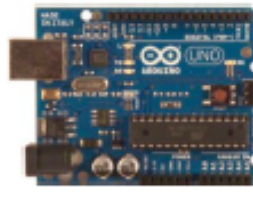


Arduino Uno R2 Front

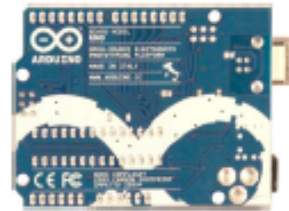


Arduino Uno SMD

Arduino Uno R3 Back



Arduino Uno Front



Arduino Uno Back

Overview

The Arduino Uno is a microcontroller board based on the ATmega328 ([datasheet](#)). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started.

The Uno differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega16U2 (Atmega8U2 up to version R2) programmed as a USB-to-serial converter.

[Revision 2](#) of the Uno board has a resistor pulling the 8U2 HWB line to ground, making it easier to put into [DFU mode](#).

[Revision 3](#) of the board has the following new features:

- 1.0 pinout: added SDA and SCL pins that are near to the AREF pin and two other new pins placed near to the RESET pin, the IOREF that allow the shields to adapt to the voltage provided from the board. In future, shields will be compatible both with the board that use the AVR, which operate with 5V and with the Arduino Due that operate with 3.3V. The second one is a not connected pin, that is reserved for future purposes.
- Stronger RESET circuit.
- Atmega 16U2 replace the 8U2.

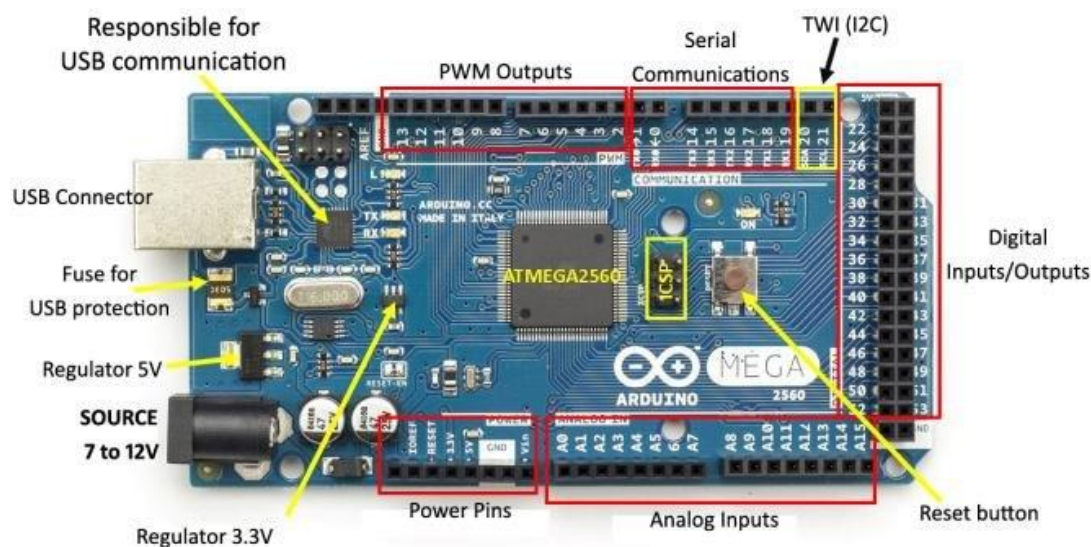
"Uno" means one in Italian and is named to mark the upcoming release of Arduino 1.0. The Uno and version 1.0 will be the reference versions of Arduino, moving forward. The Uno is the latest in a series of USB Arduino boards, and the reference model for the Arduino platform; for a comparison with previous versions, see the [index of Arduino boards](#).

Summary

Microcontroller	ATmega328
Operating Voltage	5V
Input Voltage (recommended)	7-12V

Input Voltage (limits)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
Analog Input Pins	6
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328) of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328)
EEPROM	1 KB (ATmega328)
Clock Speed	16 MHz

מיקרו בקר Arduino Mega:

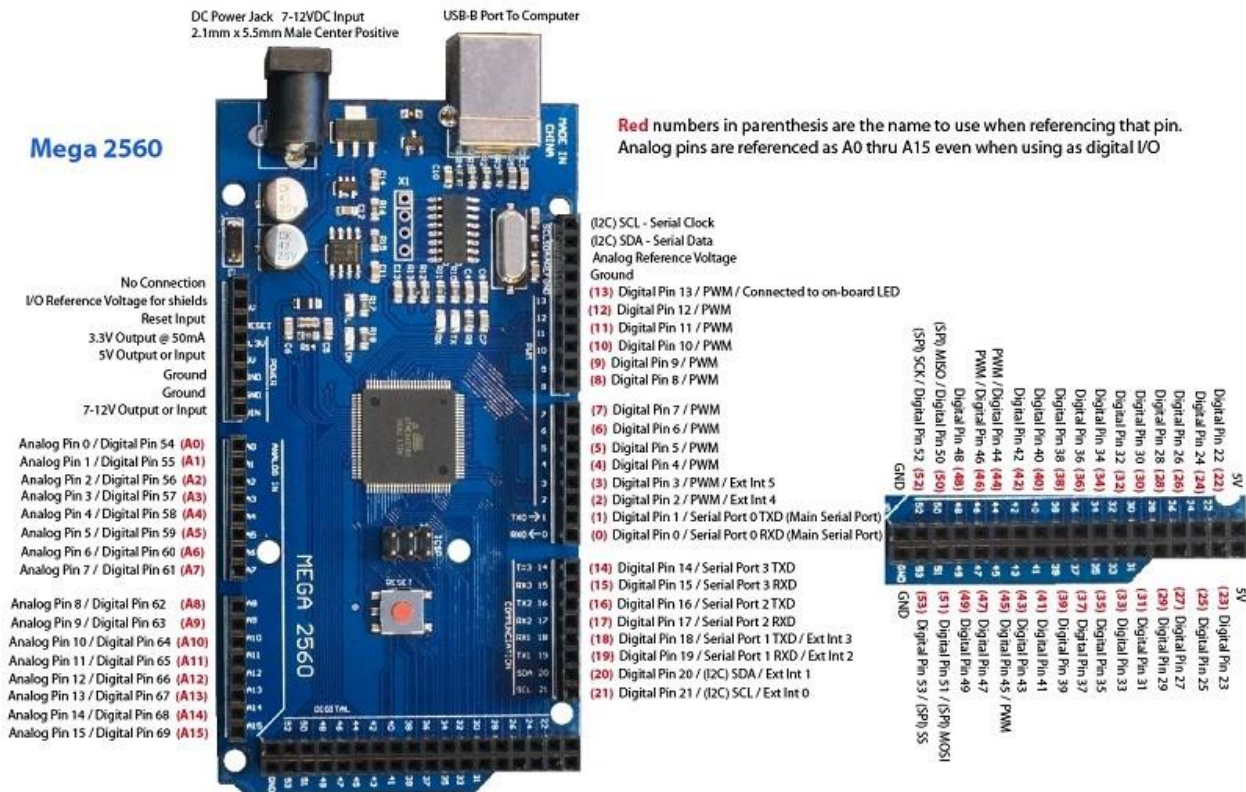


החלטתנו להשתמש בבקר ארדואינו נוסף נובעת עקב בעיית זרם. מכיוון שבמערכת ישנם מספר רכיבים אשר "גוזלים" זרם רב, נוצר מצב בלתי רצוי שבו הרכיבים לא מקבלים את הזרם הנדרש לפעולתם התקינה, אשר אמורה להיות מסופקת על ידי הארדואינו UNO.

על מנת לפתור בעיה זאת, החלטנו שמשום שכבר יש בבעלותנו ארדואינו נוסף, נשתמש בו בתור פלטפורמה נוספת למערכת וכפתרון לבעיית הזרם המוצגת לעיל, אשר תמומש בנוסף למערכת הקיימת.

מפרט טכני:

- משתמש בבקר ATmega2560.
- 54 פני I/O (דיגיטליים).
- 16 פנים אנלוגיים.
- 15 ערוצי PWM.
- זרם מקסימלי לפין הוא 40mA.
- מתח עבודה של 5V.
- 256KB של זכרון.
- 8KB זכרון SRAM.
- 4KB זכרון EEPROM.
- מהירות שעון של 16MHz.
- מתח הזנה מומלץ בין 7-12V.



The Arduino Mega 2560 is a microcontroller board based on the ATmega2560 ([datasheet](#)). It has 54 digital input/output pins (of which 14 can be used as PWM outputs), 16 analog inputs, 4 UARTs (hardware serial ports), a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. The Mega is compatible with most shields designed for the Arduino Duemilanove or Diecimila.

Schematic & Reference Design

EAGLE files: [arduino-mega2560-reference-design.zip](#)

Schematic: [arduino-mega2560-schematic.pdf](#)

Summary

Microcontroller	ATmega2560
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	54 (of which 14 provide PWM output)
Analog Input Pins	16
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	256 KB of which 8 KB used by bootloader
SRAM	8 KB
EEPROM	4 KB
Clock Speed	16 MHz

LDR 3.3.2 חיישן-אור:

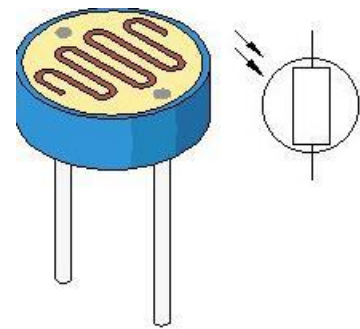
חיישן-האור הינו נגד-משתנה, אשר משתנה בהתאם לעוצמת ההארה:

עבור עוצמת הארה גבוהה, התנגדות ה-LDR שואפת ל 0,
עבור עוצמת הארה נמוכה, התנגדות ה-LDR שואפת לאינסוף.

המדידה באמצעותו אינה נעשית באופן אוטומטי על ידי החיישן, אלא על ידי המשתמש בו.

קביעת עוצמת ההארה, מצב שנע בין אור לחושך, תיעשה על ידי מדידת המתח על פני נגד-האור.

המתח על הנגד-אור משתנה עקב שינוי התנגדות נגד-אור על פי המצבים המתוארים לעיל.

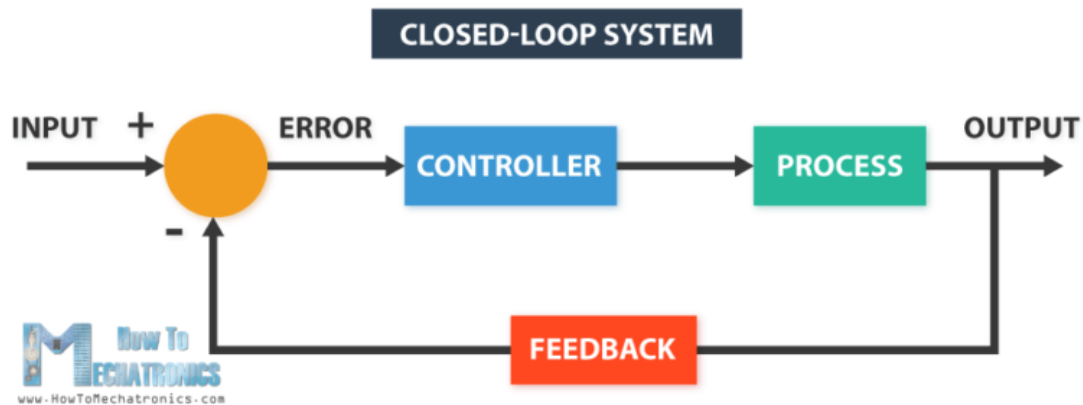


השימוש בחיישן האור מתבצע על מנת למדוד את עוצמת ההארה, וזאת בכדי לשנות את מיקום המערכת הסולארית באופן האופטימאלי כלפי מקור האור, וכן לבצע תיקוני מיקום באופן שוטף. דבר זה מתאפשר לנו על ידי מערך ארבעת חיישני-האור.

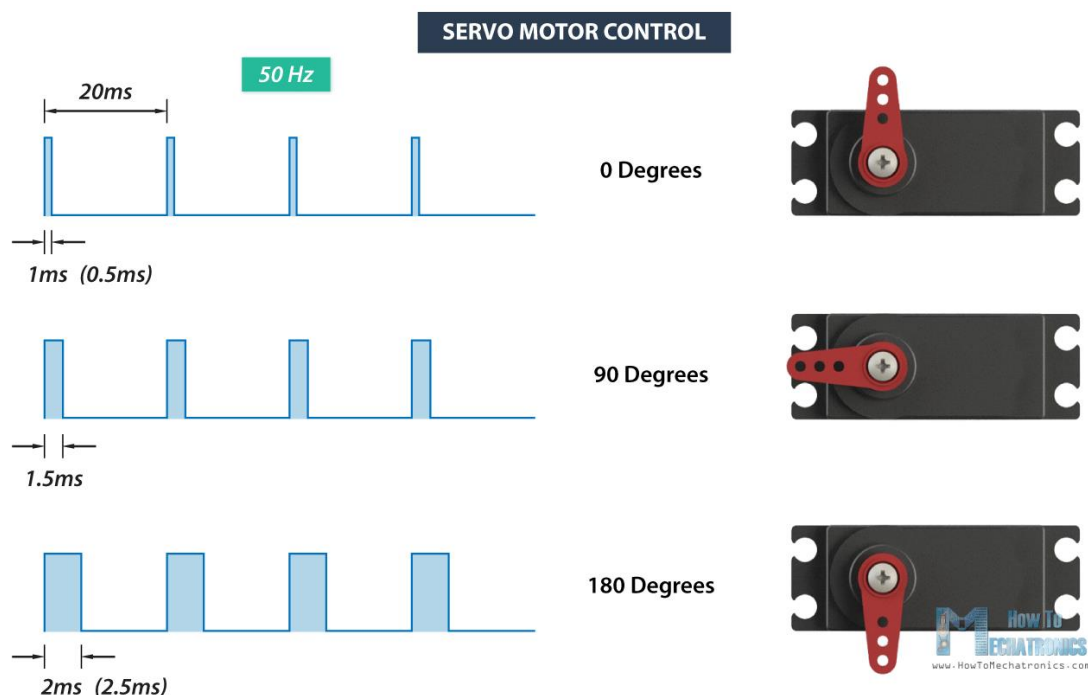
3.3.3 מנועי SERVO:

תפקיד רכיבי הסרבו הינו להזיז את הפאנלים הסולאריים אל כיוון מקור האור.

מנוע הסרבו מכיל בתוכו מערכת "סגורת לולאה", כלומר מערכת בעלת משוב.



הסרבו נשלט על ידי פולס PWM (Pulse Width Modulation). דופק זה הינו פולס חשמלי, אשר על ידי שינוי רוחב הדופק שלו, ניתן לשלוט בסרבו.



הסרבו מכיל בתוכו מנוע DC, פוטנציומטר (נגד משתנה), לוח שליטה (המכיל בתוכו את המערכת בעלת המשוב) ומערכת צירים אשר שולטים באופן הפעולה שלו.

3.3.4 מסך LCD:

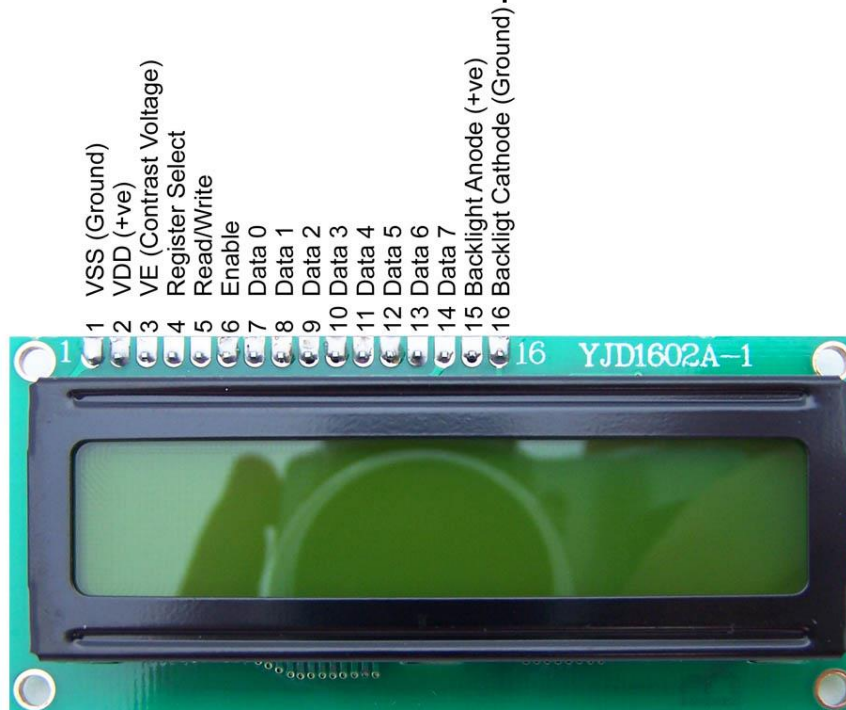
מסך LCD או בשמו המלא "Liquid Crystal Display", הוא מסך תצוגה המכיל בתוכו קריסטלים נוזליים, המשמשים אותו להצגת המידע על המסך.

יתרונותיו העיקריים של מסך ה-LCD הוא עוביו הדק וצריכת אנרגיה נמוכה (ביחס לתצוגת LED) וזאת משום שבמקום להקרין אור, מסך ה-LCD חוסם אור.

הקריסטלים הנוזליים ברכיב מציגים את המידע בעזרת תאורת רקע (Backlight).

לרכיב ה-LCD קיימים 16 חיבורים:

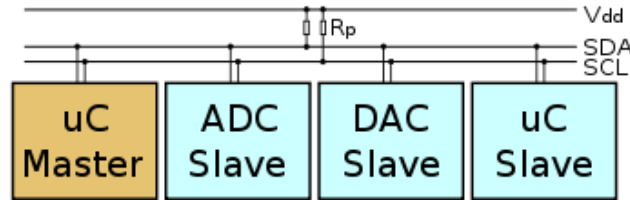
מתח, אדמה, בהירות, רגיסטר, פעולת קריאה/כתיבה, Enable (אשר אחראי על פעולת/אי פעולת הרכיב), 8 קווי מידע, אנודה וקטודה אשר אחראיים על תאורת הרקע.



קיים רכיב עזר למסך ה-LCD, (I2C) אשר תפקידו לייעל את השימוש במסך ה-LCD על ידי הקטנת מספר החיבורים מ 16 חיבורים ל 4 חיבורים בלבד.

3.3.5 I2C:

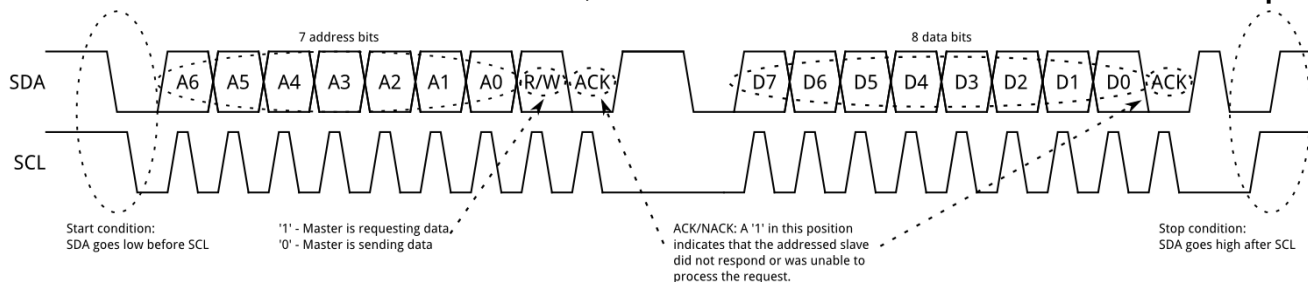
רכיב ה-I2C הינו רכיב פרוטוקול טורי-סנכרוני, בעל 2 קווי תקשורת המשמשים לתקשורת בין רכיבים בעלי מהירות נמוכה כמו: מיקרו – בקרים, ממירים A/D וממירים D/A.



ברכיב קיימים 2 קווי תקשורת: SCL אשר אחראי שכול הרכיבים יפעלו עם שעון משותף SDA שהוא קו המידע.

לרכיב ניתן לחבר מספר רב של רכיבי Slave ומספר רב של רכיבי Master. לכול רכיב Slave יש כתובת משלו, (לרוב הכתובת היא בת 7 ביטים).

רכיב ה Slave נשלט על ידי רכיב ה- Master. תקשורת I2C בסיסית משתמשת ב- 8 ביטים, לדוגמא:



העברת המידע מתחילה עם סימון התחלה (Start Condition), כאשר קו ה-SCL יורד ממתח גבוה, וקו ה-SDA יורד ממתח גבוה אחריו, קו ה-SDA משדר את כתובת רכיב ה-Slave מהביט ה-MSB לביט ה-LSB.

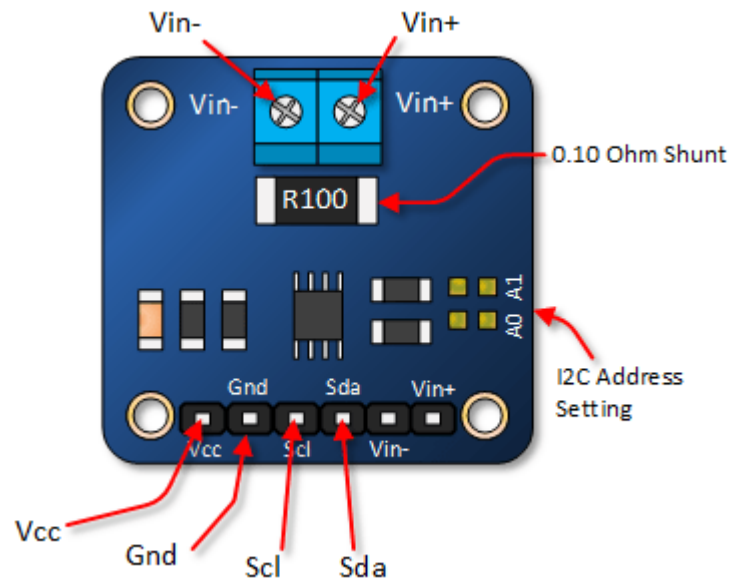
אם הביט במקום 0 של הכתובת הוא "0" לוגי, רכיב ה- Master יכתוב לרכיב, אם הוא "1" לוגי הוא יקרא מין הרכיב. לאחר ביט הקריאה/כתיבה מגיע ביט ACK/NACK, אשר במקרה שהוא "1" לוגי, דבר זה אומר כי רכיב ה-Slave אינו מגיב או לא מצליח לעבד את הבקשה.

לאחר מכן המידע נכתב לרכיב או נקרא מן הרכיב ולבסוף משודר תנאי סיום (Stop Condition) אשר ממומש על ידי עליה של SDA לאחר עליה של SCL.

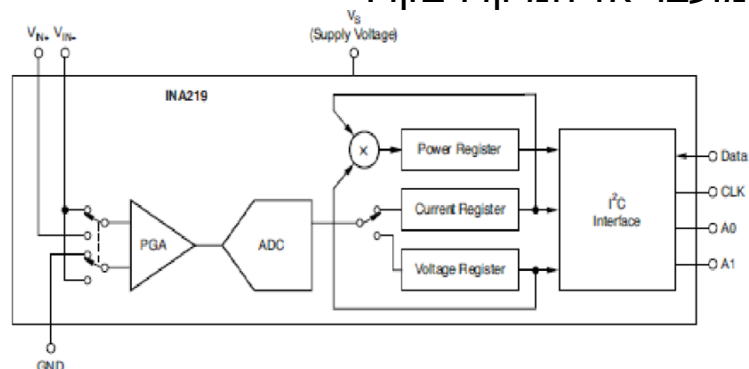
INA219 3.3.6:

רכיב ה-INA219 הינו חיישן שתפקידו לקרוא את המתח אשר מתפתח על נגד בעל התנגדות מאוד קטנה (לרוב 0.1Ω) אשר נקרא Shunt.

כמו כן הרכיב קורא גם את המתח הנכנס לרכיב, ובעזרת חישוב מתמטי מציג נתונים של מתח, זרם והספק העומס הנמדד.

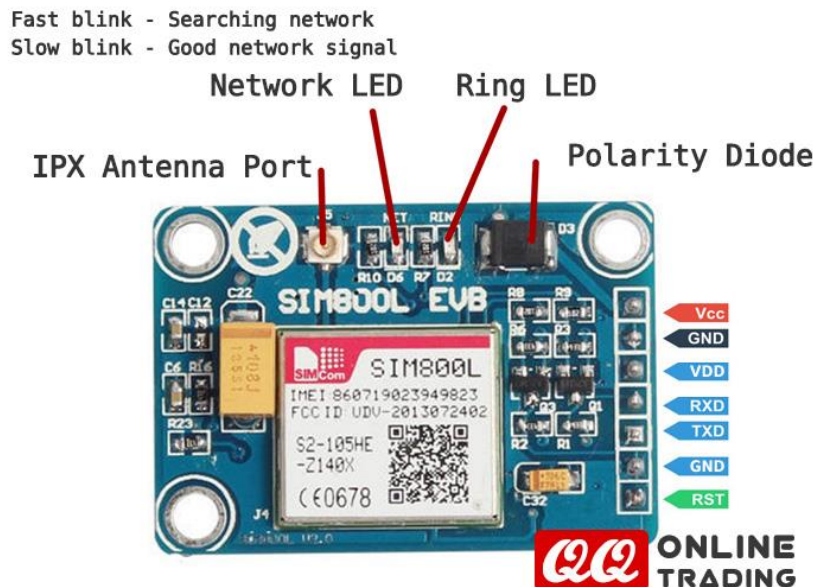


פעולות המדידה מתבצעות באופן הבא:
ברכיב קיים מגבר מדויק (PGA), אשר מודד את המתח על פני הנגד ומגביר את המתח (זאת על מנת לקרוא מתחים קטנים).
לאחר פעולת ההגברה, המתח המוגבר מועבר לרכיב A/D, כלומר ממיר מאנלוגי לדיגיטלי בעל 12 ביטים.
המידע הספרתי עובר אל רכיב אשר מחשב מתח ואל רכיב אשר מחשב זרם, התוצאות שלהם עוברים דרך ערבל ומועברות אל רכיב אשר מחשב את ההספק.
לאחר פעולות החישוב לעיל, מידע ספרתי זה עובר אל רכיב ה-I2C ומועבר אל המיקרו-בקר.



sim800l evb 3.3.7:

רכיב ה-sim800l הינו רכיב תקשורת, המשמש אותנו לתקשורת בין המיקרו-בקר לבין המכשיר הסלולרי (למשתמש).
רכיב ה-sim800l מתקשר עם המכשיר הסלולרי בעזרת רכיב אשר כלול בו, הנקרא מודול GSM.



מודול ה-GSM או השמו המלא "Global System For Mobile Communication", הינו מודם תקשורת סלולרית. מודול ה-GSM ממיר לנו אות אנלוגי (מידע אשר נקלט על ידי), לאות דיגיטלי (מידע שעובר אל הרכיב), ולהיפך. המידע המשודר מרכיב ה-GSM עובר תהליך ריבוב הנקרא ריבוב TDM (Time Division Multiplexing). תהליך ה-TDM הינו תהליך אשר בו רכיב ה-GSM מגדיר טווח זמן מסוים למשתמשים שונים, על אותו תדר. כתוצאה מפעולה זו, ניתן להכניס מידע רב יותר על תדר מידע אחד, משום שעד שהמידע נדגם שנית מן אות המידע, קיים "זמן מת", כלומר זמן שבו ניתן לשנות את המידע ושצרכן נוסף ידגום את מידע זה.

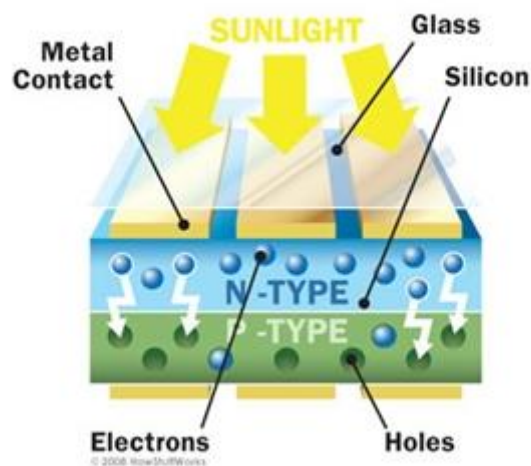
המידע המשודר מן הרכיב או נקלט אל הרכיב עובר דרך חיבורי ה-RX, TX אלו הם חיבורים של שידור מידע (Transmit) או קליטת מידע (Receive). חיבורים אלה פועלים באופן טורי, כלומר ביט אחרי ביט של המידע-עובר בקו.

3.3.8 פאנלים סולאריים:

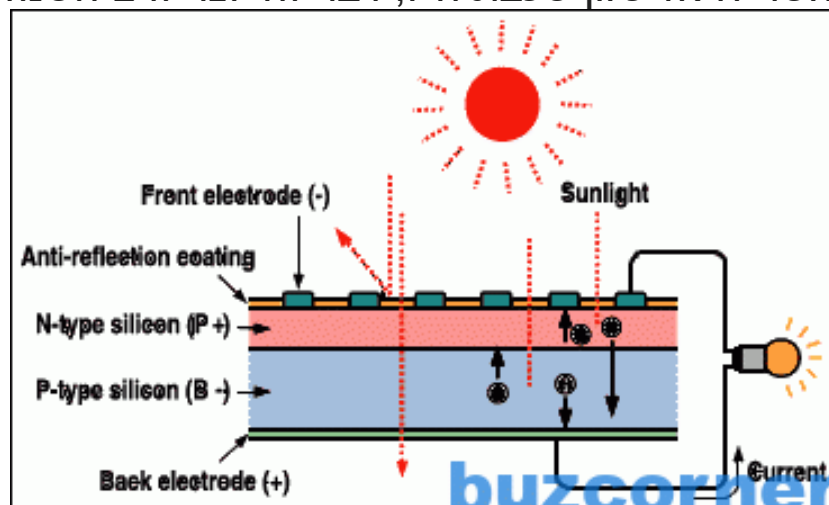
תפקידם של הלוחות הסולאריים הינו המרת אנרגיה סולארית לאנרגיה חשמלית.

המרה זו מתבצעת על ידי כך, כאשר הפוטונים (חלקיקי אור) מצליחים "לשחרר" אלקטרונים מהאטומים שלהם, ובכך מאפשרים זרימת אלקטרונים, פעולה זו הינה יצירת החשמל. הלוחות מורכבים מתת-יחידות קטנות יותר הנקראות תאים פוטו-וולטאים.

תאים אלה מורכבים משתי שכבות של מוליכים למחצה, שכבה אחת מסוג N, כלומר שכבה בעלת עודף אלקטרונים, ושכבה אחת מסוג P, כלומר שכבה בעלת חוסר אלקטרונים.

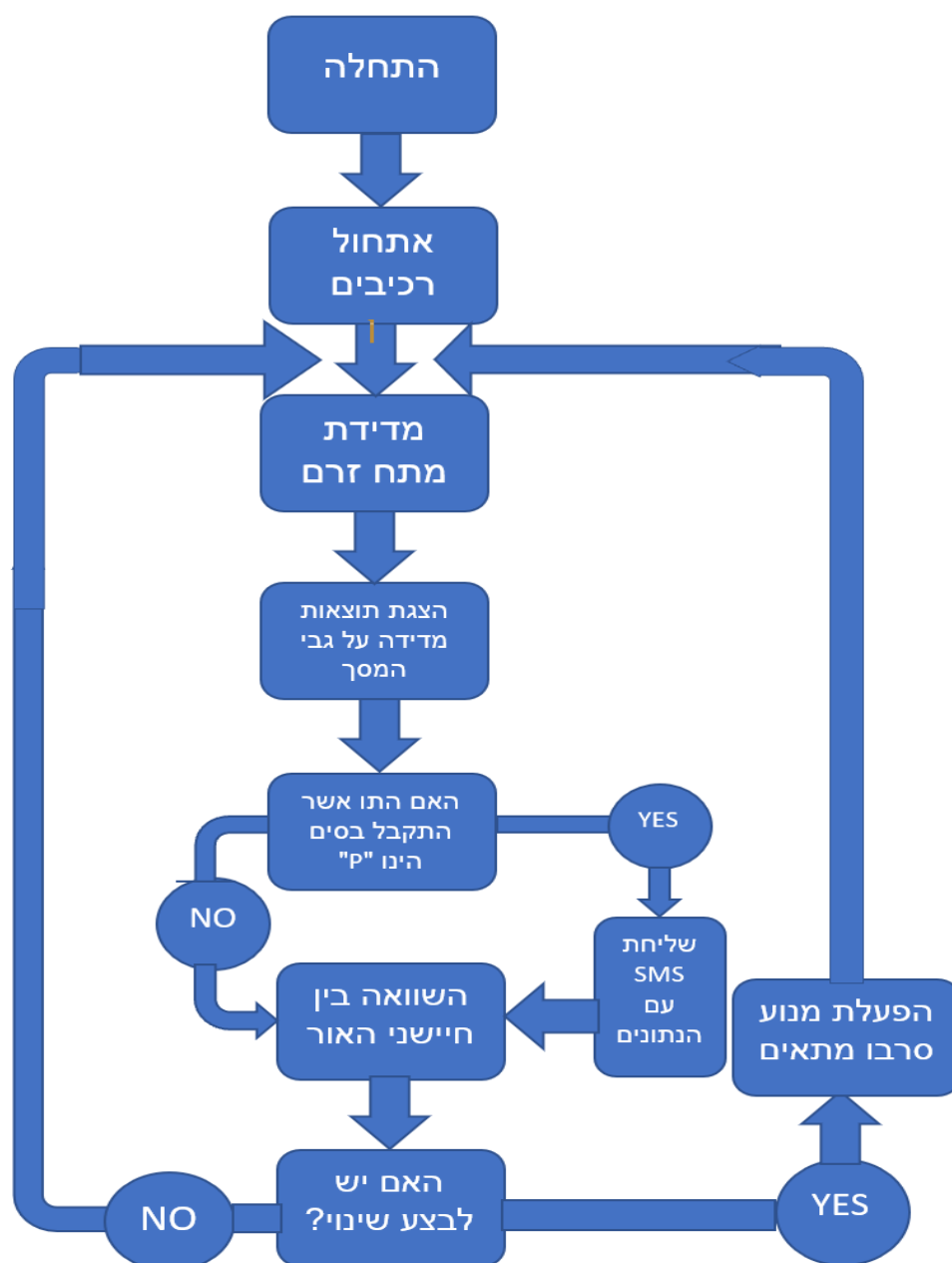


שתי שכבות אלו יוצרות בניהם צומת PN שלא מאפשרת מעבר אלקטרונים, אזור זה נקרא אזור המחסור. כאשר פוטון בעל אנרגיה מספקת פוגע בשכבת הN הוא משחרר אלקטרון אשר זז אל כיוון שכבת הP, דבר זה יוצר זרם חשמלי.



פרק 4 – תכנה

תרשים זרימה כלל מערכת:



4.1 אלגוריתמים של תתי שגרות

4.1.1 תת – שגרה מדידת אחוז סוללה

```
int batread = A2;
```

```
float BAT = 0; תת שגרה זו הינה פעולה האחראית על מדידת אחוז הסוללה.
```

```
void setup() {
```

```
    Serial.begin(9600);
```

```
    pinMode(batread, INPUT);
```

```
}
```

```
void loop() {
```

```
    BAT = analogRead(batread) * 0.00488
```

```
    BAT -= 3.7;
```

```
    BAT = BAT / 0.0024;
```

```
}
```

לביצוע הפעולה המערכת לוקחת קריאה אנלוגית

מן הפורט האנלוגי- A2 , לאחר מכן

מכפילה את הקריאה במספר שמכיל בתוכו

גם את פעולת ההמרה ממספר "ספרתי" לערך אנלוגי,

וכמו כן המרה של הערך האנלוגי לאחוז מן הסוללה

נקרא למדידה מן הסוללה N:

$$V_{in} = V_{lsb} * N$$

זוהי המשוואה הבסיסית של כול ממיר אנלוגי לדיגיטלי.

$$V_{lsb} = \frac{V_{inmax}}{N_{max}} = \frac{5}{2^{10} - 1} = 0.00488$$

במשוואה לעיל אנו מחשבים את הרזולוציה של

הארדואינו על ידי חלוקת המתח המירבי שהארדואינו

יכול לקרוא במספר ה"ספרתי" המירבי שלו, המספר הינו 1023.

הארדואינו עובד במערכת של 10 ביט, אנו מחסרים 1 משום שהרמה

הראשונית היא 0.

$$V_{in} = N * 0.00488$$

לאחר קבלת המתח הנמדד אנו רוצים להעביר את ערך זה לאחוזים.

משום שהמערכת עובדת בסקאלה של 3.7V עד 4.2V לפי נתונים

"יבשים" של סוללת הליפון, אנו צריכים לראות עבור איזו "קפיצה" של

מתח אנו עולים באחוז אחד.

מדדנו את הסוללה לאחר שהיא טעונה במלואה וראינו כי המתח

שמתפתח עליה הוא 3.94V ולפיכך:

$$3.94V = 100\%$$

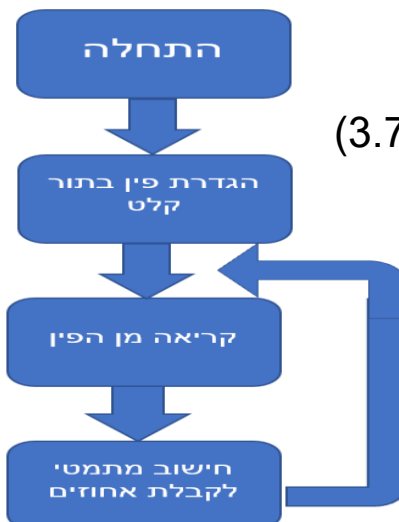
$$3.7V = 0\%$$

נחסר את הערך העליון (3.94V) מהערך התחתון (3.7V)

ונחלק ב-100 על מנת לקבל את ה"רזולוציה":

$$3.94V - 3.7V = 0.24$$

$$\frac{0.24}{100} = 2.4mV = 1\%$$



```

#include <Wire.h>
#include <LiquidCrystal_I2C.h>

LiquidCrystal_I2C lcd(0x27,16,2);

int current_mA = 0;
int loadvoltage = 0;
int power_mW = 0;
int BAT = 0;

void setup() {
  lcd.init();
  lcd.backlight();
}

void loop() {
  lcd.setCursor(0, 0);
  lcd.print("I: ");
  lcd.print(current_mA);
  lcd.setCursor(10, 0);
  lcd.print("V: ");
  lcd.print(loadvoltage);
  lcd.setCursor(0, 1);
  lcd.print("P: ");
  lcd.print(power_mW);
  lcd.setCursor(8, 1);
  lcd.print("BAT:");
  lcd.print(BAT);
  lcd.print("%");
}

```

4.1.2 תת – שגרה תצוגת מסך LCD

בתת - שגרה זו המטרה שלנו היא להפעיל את מסך ה LCD ולהציג עליו את הנתונים אשר אנו נקבל משאר הרכיבים במעגל. נתונים כגון:

מתח/זרם המגיעים מן הלוחות הסולאריים, ההספק של הלוחות ואחוז טעינת הסוללה הנטענת. אנו משתמשים בספריית ה WIRE המאפשרת לנו להשתמש ברכיב I2C המחובר אל מסך ה LCD. כמו כן אנו משתמשים בספריית המסך על מנת להמיר את המידע אשר אנו רוצים להציג אליו למידע אשר יוכל להיות מוצג על המסך. כעת מוגדרת כתובת מסך ה LCD על מנת שיהיה ניתן לתקשר עימו דרך רכיב ה I2C והגדרת גודל המסך.

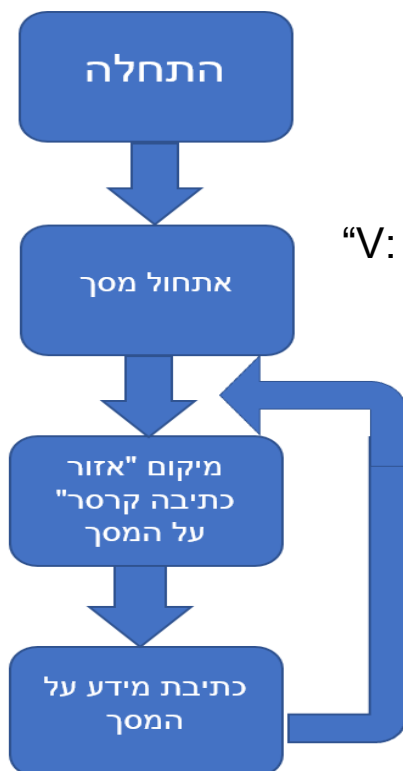
לאחר מכן אנו מדליקים את מסך ה LCD כולל תאורת הרקע שלו.

על ידי שימוש בלולאה אנו נדפיס במחזוריות את הנתונים על גבי המסך.

שורת הקוד lcd.setCursor מאפשרת לנו לבחור באיזה אזור במסך ה LCD אנו רוצים להשתמש, אנו משתמשים ב LCD 16x2 ולכן יהיו לנו 2 שורות ו 16 טורים בשימוש.

המערכת מדפיסה את שם המשתנה לדוגמא: "V: " ולאחר מכן את המשתנה.

דבר זה נובע עקב מגבלות תוכנה.



```
#include <SoftwareSerial.h>

SoftwareSerial sim(2, 3);

int _timeout;
String _buffer;
String number = "מספר פלאפון רצוי";
int current = 0;
int power = 0;
int voltage = 0;

void setup() {
    _buffer.reserve(50);
    sim.begin(9600);
    sim.println("AT+CNMI=2,2,0,0,0");
    delay(100);
}

void loop() {
    if (sim.available() > 0) {
        if (sim.read() == 'P')
            SendMessage();
    }
}

void SendMessage()
{
    sim.println("AT+CMGF=1");
    delay(100);
    sim.println("AT+CMGS=\"" + number + "\"");
    delay(100);
    String SMS = ("I: ");
    SMS += (current);
    SMS += (" V: ");
    SMS += (voltage);
    SMS += (" P: ");
    SMS += (power);
    Serial.println(SMS);
    sim.println(SMS);
    delay(100);
    sim.println((char)26);
    delay(100);
    _buffer = _readSerial();
}

String _readSerial() {
    _timeout = 0;
    while (!sim.available() && _timeout < 12000 )
    {
        delay(13);
        _timeout++;
    }
    if (sim.available()) {
        return sim.readString();
    }
}
```

4.1.3 תת - שגרה רכיב GSM

בתת - שגרה זו המטרה שלנו היא להפעיל את רכיב התקשורת-GSM.

כאשר הרכיב מקבל הודעה עם התו "P", הוא ישלח לנו את נתוני המערכת (מתח, זרם והספק המערכת).

אנו משתמשים בספרית תקשורת המאפשרת לנו לשלוח ולקבל מידע דרך כניסות RX וTX של המיקרו-בקר. אנו מגדירים לבקר אילו כניסות אנו נשתמש בשביל ה-RX וה-TX (ה-TX הוא הראשון שנרשם כלומר "2").

שורת ה-buffer.reserve משמשת אותנו לזמן ההשהיה של הרכיב, כלומר זהו הזמן שהרכיב מקבל "זמן למנוחה" בין ביצוע פעולה אחת לשנייה.

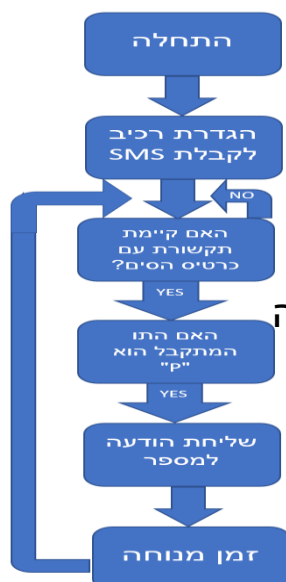
לאחר מכן אנו מאתחלים את כרטיס הסיים ומפעילים אותו בקצב של 9,600 bps, כלומר מהירות קריאה של 9,600 ביטים לשנייה.

לאחר פעולה זו המערכת תשלח לרכיב פקודה אשר תגרום לו "לקרוא" את ההודעות המתקבלות על הסיים שהוכנס.

בלולאה קיים תנאי, רק אם הסיים יקבל הודעה או תקשורת כלשהי ובנוסף תתקבל הודעה עם התו "P", המיקרו-בקר יכנס לפונקציה חיצונית שתממש את שליחת ההודעה עם הנתונים.

בפונקציית שליחת ההודעה, מוגדר לרכיב לעבור למצב שליחת הודעה ומוגדר לו המספר פלאפון אליו הוא צריך לשלוח את ההודעה. כעת אנו נרכיב את ההודעה אשר אנו רוצים לשלוח, אנו נחבר אל ההודעה את התווים הקבועים, ביחד עם המשתנים אשר אנו רוצים שיופיעו

בתוך גוף ההודעה. לאחר בניית כול ההודעה נשלח לרכיב GSM פקודה לשלוח את ההודעה הנ"ל למספר הפלאפון המוגדר במערכת. כמו כן נשלח לרכיב GSM התו "26" זהו בעצם תו בקוד ASCII אשר מתורגם ל-"Ctrl+Z", תו זה מסמל את סוף ההודעה ומודיע לרכיב לשלוח את ההודעה. כעת ניתן לרכיב זמן השהייה מסוים על מנת לא "להעמיס" עליו ועל מנת שיוכל לבצע את הפקודה אשר נשלחה אליו במלואה.



```
#include <Servo.h>
```

```
Servo leftright;
Servo updown;
int topleft;
int topright;
int downleft;
int downright;
int horizontal = 90;
int vertical = 100;
int waittime = 30;
```

```
void setup() {
  Serial.begin(9600);
  leftright.attach(9);
  updown.attach(10);
  leftright.write(horizontal);
  updown.write(vertical);
}
```

```
void moveright() {
  horizontal++;
  leftright.write(horizontal);
}
```

```
void moveleft() {
  horizontal--;
  leftright.write(horizontal);
}
```

```
void movedown() {
  vertical--;
  updown.write(vertical);
}
```

```
void moveup() {
  vertical++;
  updown.write(vertical);
}
```

```
void loop() {
  topleft = analogRead(A2);
  topright = analogRead(A1);
  downleft = analogRead(A3);
  downright = analogRead(A0);
```

```
  if (horizontal > 180)
    horizontal = 180;
  if (horizontal < 0)
    horizontal = 0;
  if (vertical > 150)
    vertical = 150;
  if (vertical < 60)
    vertical = 60;
```

```
  if ((downright < topright) or (downleft < topleft)) {
    movedown();
    delay(waittime);
  }
```

```
  if ((downright > topright) or (downleft > topleft)) {
    moveup();
    delay(waittime);
  }
```

```
  if ((downright < downleft) or (topright < topleft)) {
    moveright();
    delay(waittime);
  }
```

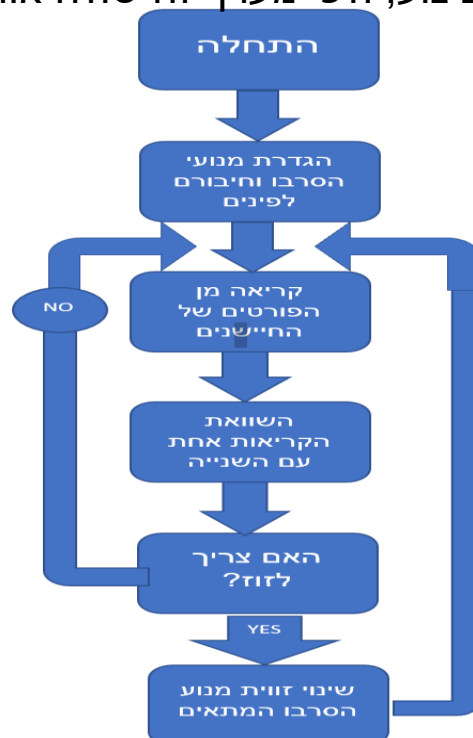
```
  if ((topright > topleft) or (downright > downleft)) {
    moveleft();
    delay(waittime);
  }
}
```

4.1.4 תת - שגרה זיהוי אור ותזוזה אליו

תת - שגרה זו שולטת לנו במנועי הסרבו ובמערך נגדי האור של המערכת, אשר תפקידם לזהות את מיקום מקור האור. לביצוע תת שגרה זו נשתמש בספריית ה-SERVO אשר מאפשרת לנו להוציא למנועי הסרבו אות PWM אשר המנועים יוכלו לפעול לפיו.

בתת השגרה, הגדרנו את הפינים אשר מחוברים לנו לרגלי ה-PWM של מנועי הסרבו ו"חיברנו" אותם לכל מנוע סרבו בנפרד. כעת המערכת תאפס אותם למצב ההתחלתי שלנו. לאחר מכן אנו מגדירים מספר פונקציות שהמנועים יפעלו לפיהם בתלות בקריאה ממערך נגדי האור. בלולאה, המיקרו-בקר לוקח קריאות מהרגליים האנלוגיים, אשר מחוברים אליהם נגדי האור.

בעזרת הקוד המערכת תבצע "תיקונים" לקריאות משום שלכול נגד אור יש אחוז סטיה מסוים וגם למערכת אין צורך בדיוק כזה גדול שהקריאה מספקת לה. כעת אנו מונועים מהזווית אשר נשלחת למנועי הסרבו לעלות מעל ל-180 מעלות או לרדת מתחת ל-0 מעלות על מנת לא לפגוע במנוע הסרבו האופקי, ומגבילים את זווית התזוזה של המנוע האנכי בגלל מגבלות פיזיות כגון: רכיבי מבנה המפריעים לתזוזתו. כעת המיקרו-בקר עושה השוואה עם הקריאות של נגדי האור ובודק איזו תנועה צריך לשלוח למנועי לביצוע, ולפי מערך זה שולח אות לביצוע הפעולה.



4.1.5 תת – שגרה מדידת מתח/זרם

```
#include <Wire.h>
#include <Adafruit_INA219.h>
```

```
Adafruit_INA219 ina219;
```

```
float shuntvoltage = 0;
float busvoltage = 0;
float current_mA = 0;
float loadvoltage = 0;
float power_mW = 0;
```

```
void setup() {
  Serial.begin(115200);
  uint32_t currentFrequency;
  ina219.begin();
}
```

```
void loop() {
  shuntvoltage = ina219.getShuntVoltage_mV();
  busvoltage = ina219.getBusVoltage_V();
  current_mA = ina219.getCurrent_mA();
  power_mW = ina219.getPower_mW();
  loadvoltage = busvoltage + (shuntvoltage / 1000);
}
```

הרכיב לתת שגרה זו משתמש ברכיב I2C ,

אנו צריכים להשתמש בספריה התאפשר תקשורת עם הרכיב.

כמו כן אנו משתמשים גם בספריה של הרכיב עצמו.

ספריה זו מאפשרת שימוש בפונקציות פנימיות שלה.

כעת מוגדר לתוכנה כי אנו משתמשים ברכיב INA219

וכמו כן ניתן להגדיר את כתובת הרכיב. במידה ולא מוגדרת

כתובת, כתובת ברירת המחדל של הרכיב היא

40H.

כעת אנו מגדירים את תדר הבדיקה בגודל

של 32 ביט

כעת המערכת תאתחל את הפעולה

של הרכיב.

בלולאה אנו משתמשים בפקודות פנימיות

של הספרייה, שמטרתה להכניס את המדידות של הרכיב למשתנים

אשר נוכל לעשות בהם פעולות במידת הצורך.

קיימות מדידות כגון:

המתח על השאנט (נגד של 0.1Ω) ב mV,

מתח בין ההדקים של הרכיב ב V,

הזרם דרך הרכיב ב mA,

ההספק שעובר דרך הרכיב ב mW

והחישוב של מתח העומס המחובר, אשר הינו המתח ביציאת רכיב

המדידה וחיבור של המתח על השאנט ב V,

שעל מנת להמיר מ mV ל V מחלקים ב-1000.



4.2 קובץ LIST של התוכנה

ארדואינו UNO (תוכנת התקשורת והצגת נתונים ומדידת נתונים):

```
#include <Wire.h>
#include <SoftwareSerial.h>
#include <LiquidCrystal_I2C.h>
#include <Adafruit_INA219.h>
Adafruit_INA219 ina219;
SoftwareSerial sim(2, 3);
LiquidCrystal_I2C lcd(0x27, 16, 2);

String _buffer;
int _timeout;
String number = "+972507168509";
float shuntvoltage = 0;
float busvoltage = 0;
float current_mA = 0;
float loadvoltage = 0;
float power_mW = 0;
int current = 0;
int voltage = 0;
int power = 0;
int BAT = 0;
int batread = A2;

void setup() {
  _buffer.reserve(50);
  sim.begin(115200);
  sim.println("AT+CNMI=2,2,0,0,0");
  uint32_t currentFrequency;
  ina219.begin();
  lcd.init();
  pinMode(batread, INPUT);
}

void SendMessage() {
  sim.println("AT+CMGF=1");
  delay(100);
  sim.println("AT+CMGS=\"" + number + "\"\r");
  delay(100);
  String SMS = ("I: ");
  SMS += (current_mA);
  SMS += (" V: ");
  SMS += (loadvoltage);
  SMS += (" P: ");
  SMS += (power_mW);
  SMS += (" BAT: ");
  SMS += (BAT);
  SMS += ("%");
  Serial.println(SMS);
  sim.println(SMS);
  sim.println((char)26);
  delay(100);
}
```

```

void LCD() {
    BAT = analogRead(batread) * 0.00488;
    BAT -= 3.7;
    BAT = BAT / 0.0024;
    lcd.backlight();
    lcd.setCursor(0, 0);
    lcd.print("I: ");
    lcd.print(current_mA);
    lcd.setCursor(10, 0);
    lcd.print("V: ");
    lcd.print(loadvoltage);
    lcd.setCursor(0, 1);
    lcd.print("P: ");
    lcd.print(power_mW);
    lcd.setCursor(8, 1);
    lcd.print("BAT:");
    lcd.print(BAT);
    lcd.print("%");
}

void measurement() {
    shuntvoltage = ina219.getShuntVoltage_mV();
    busvoltage = ina219.getBusVoltage_V();
    current_mA = ina219.getCurrent_mA();
    power_mW = ina219.getPower_mW();
    loadvoltage = busvoltage + (shuntvoltage / 1000);
}

void loop() {
    measurement();
    LCD();
    if (sim.read() == 'P') {
        SendMessage();
    }
}

```

ארדואינו MEGA (תוכנת תזוזת המנועים וזיהוי מיקום מקור האור):

```
#include <Servo.h>

Servo leftright;
Servo updown;
int topleft;
int topright;
int downleft;
int downright;
int horizontal = 90;
int vertical = 100;
int waittime = 30;

void setup() {
  Serial.begin(9600);
  leftright.attach(9);
  updown.attach(10);
  leftright.write(horizontal);
  updown.write(vertical);
}

void moveright() {
  horizontal++;
  leftright.write(horizontal);
}

void moveleft() {
  horizontal--;
  leftright.write(horizontal);
}

void movedown() {
  vertical--;
  updown.write(vertical);
}

void moveup() {
  vertical++;
  updown.write(vertical);
}
```

```

void loop() {
  topleft = analogRead(A2);
  topright = analogRead(A1);
  downleft = analogRead(A3);
  downright = analogRead(A0);

  if (horizontal > 180)
    horizontal = 180;
  if (horizontal < 0)
    horizontal = 0;
  if (vertical > 150)
    vertical = 150;
  if (vertical < 60)
    vertical = 60;

  if ((downright < topright) or (downleft < topleft)) {
    movedown();
    delay(waittime);
  }

  if ((downright > topright) or (downleft > topleft)) {
    moveup();
    delay(waittime);
  }

  if ((downright < downleft) or (topright < topleft)) {
    moveright();
    delay(waittime);
  }

  if ((topright > topleft) or (downright > downleft)) {
    moveleft();
    delay(waittime);
  }
}

```

פרק 5 – סיכום ומסקנות

5.1 מדידות ותוצאות

במסגרת פרויקט זה, פיתחנו מערכת סולארית המסוגלת לעקוב אחר מקור אור, המשלבת תקשורת כמסגרת המקצוע. העבודה על פרויקט זה הייתה מאתגרת אך מעניינת מאוד. מספר פעמים במהלך העבודה על פרויקט זה, נאלצנו להתמודד עם תקלות אשר היוו מכשול בדרך אך הצלחנו למצוא להן פתרונות. העבודה על פרויקט זה דרשה חקר במכלול הנושאים, למידת הרכיבים ושימוש בהם, הכרת המיקרו בקרים בהם אנו השתמשנו, ושימוש בצידוד ייעודי. במהלך פרויקט זה שיפרנו את יכולות העבודה שלנו בשלל התחומים תוך התבססות על ידע אשר נרכש במסגרת הלימודים הסטנדרטיים במכללה ושימוש באמצעים נוספים כגון התייעצות עם מרצים, ספרות ואינטרנט. כמו שנאמר, נותר למצוא חומרים רבים באינטרנט, אך זהו אינו הפתרון הסופי, כלומר כל שימוש בחומרים הקיימים במקורות שונים בהם השתמשנו, הועבר והותאם לצורך בו אנו השתמשנו. התאמתו לתוכנית סופית, הרכבת מעגלים פיסית, הבנה ושינוי של תוכניות וכדומה דרשה ישיבה מרובה ושעות רבות של הבנה של פעולת הרכיבים והתוכניות להם.

5.1.1 רכיב SIM800:

בהקשר של רכיב התקשורת, אשר עימו נאלצנו להתעכב ולהקדיש כוחות מאומצים, מהסיבה שרכיב ה-GSM הינו רכיב מורכב. לאחר מחקר ממושך על הרכיב, הבנו כי הרכיב עובד רק עם חברות מסוימות (סלקום ואורנג' אשר תומכות בדור 2G). חיבורו עם רכיבים נוספים אשר לא אפשרו פעולה תקינה של הרכיב ולכן נדרש היה לחבר אותו למקור כוח חיצוני. שלל קשיים אלו ועוד נוספים ציינו במחברת עבודה שלנו. כמו כן נקלענו לבעיה הכי גדולה שלו עם הרכיב שהייתה קצב הפעולה שלו ביחד עם שאר הרכיבים, משום שרכיב הזרם והמתח שעימו אנו משתמשים, עבד בקצב של 115200bps בעוד והפעלנו את רכיב ה-GSM שיעבוד בקצב של 9600bps דבר זה גרם להרבה בעיות עם רכיב התקשורת ודרש מאיתנו זמן רב עד שפתרנו זאת ע"י חיבור הרכיב למקור כוח חיצוני ושינוי קצב העבודה שלו.

בדיקות רכיב הסיים:

בתחילת העבודה השתמשנו בכרטיס הסיים של חברת "פלאפון" ושמנו לב כי הרכיב עובד אך לא נשלחת או מתקבלת הודעה על ידי הרכיב והוא אינו מתחבר לרשת. בעקבות הפקודה שהצלחנו למצוא שמזהה אילו רשתות הרכיב מצליח לעבוד איתן, הבנו כי הרכיב עובד רק עם חברות סלקום ואורנג'.

```
AT+COPS=?  
Call Ready  
  
SMS Ready  
  
+COPS: (1,"Orange","ORANGE","42501"),(1,"Cellcom","Cellcom","
```

על מנת לראות האם הרכיב אכן מקבל הודעות או שולח הודעות השתמשנו במסך הסריאלי של תוכנת הארדואינו ובמכשיר סלולארי שיאמתו לנו שאכן נשלחה או התקבלה הודעה. כמו כן, לשם הכרת פעולות הרכיב ומגבלותיו, בדקנו את האפשרות של חיוג אל מכשיר סלולארי או קבלת שיחה, למרות שאין לנו הצורך בשימוש פעולות אלה בפרויקט זה.

על פי דפי היצרן ראינו כי הרכיב עובד עם זרם מקסימלי בעל עוצמה של 2A במהלך השידור ולכן למרות שתכננו לחבר את הרכיב ישירות לארדואינו, הבחנו כי כאשר אנו מחברים את הרכיב בחיבור ישיר, הרכיב אינו עובד כפי שצריך ולכן חיברנו אותו למקור כוח חיצוני.

4.1. Power Supply

The power supply range of SIM800L is from 3.4V to 4.4V. Recommended voltage is 4.0V. The transmitting burst will cause voltage drop and the power supply must be able to provide sufficient current up to 2A. For the VBAT

על מנת לוודא שהקוד נכון ואכן מגיב כאשר מקבלת ההודעה עם התו "P", שלחנו מספר הודעות עם אותיות שונות ומספרים בכדי לראות שהרכיב אכן לא שולח הודעה, לאחר מכן שלחנו לו את ההודעה הרצויה "P" וראינו כי הוא מגיב ואכן שולח לנו את הנתונים הנדרשים.

5.1.2 מנועי SERVO:

בתכנון המערכת הכוללת הראשונית רצינו להרכיב על גבי מנועי הסרבו 2 פאנלים סולאריים אשר יאפשרו לנו טעינה מהירה יותר של הסוללה בעזרת הוספת זרם מהפאנל הנוסף, כאשר חיברנו את המערך למנועים ראינו כי המנועים מתקשים להחזיק את הפאנלים ואם בכלל לזוז, המערך היה כבד עד כדי כך שנהרסו לנו 2 המנועים הראשוניים והחלטנו להוריד לוח אחד על מנת שהם יוכלו להזיז אותו. לאחר שהסרנו מן המערך את הפאנל הסולארי השני ונשארו עם פאנל בודד ונאלצנו להחליף את מנועי הסרבו עקב התקלקלותם קודם, המערך חזר לתפקד כראוי.

658	797	708	763
659	798	708	763
658	798	708	763
659	797	708	763
658	797	708	763
658	797	708	763
659	797	709	763
658	797	709	763
658	797	708	763
658	798	708	763
658	797	708	763
659	798	709	763
658	797	708	763
659	798	708	763
658	797	708	763
658	797	708	763
659	797	709	763

5.1.3 מערך זיהוי מקור אור:

בתכנון המערכת הכוללת הראשונית הנחנו כי נגדי האור זהים לגמרי, אך לאחר בדיקה ראשונית ראינו כי המערך שלנו לא עובד והחלטנו לבדוק האם הנגדים אכן זהים על ידי הארה של מקור אור ישירות על כולם בצורה זהה ומדידה שלהם במסך הסיריאלי של תוכנת הארדואינו. במהלך בדיקה זו ראינו כי הנגדים קצת שונים זה מזה ולכן החלטנו לעשות תיקונים למדידה שלהם מצד התוכנה.

5.1.4 ספר פרויקט:

כתיבת הספר דרשה גם היא יכולות כתיבה של ספר שיכיל היקף מידע מספיק אודות האלמנטים הרבים של הפרויקט והתיעוד של הפרויקט עצמו. כתיבת ספר בממדים כאלו היה משהו שטרם נדרשנו לעשות, מאיפה להתחיל? איך לכתוב? איך לסדר את הספר? מה להכניס אליו? כמובן בעזרת המנחה שלנו קיבלנו קו כתיבה לעבוד איתו אך עדיין, הכתיבה דרשה ישיבה רבה על הספר, עם כל החומרים שצברנו במהלך הפרויקט ואיגוד כולם בתוך הספר בפורמט מתאים וראוי. אנו מקווים שאת הכתיבה עשינו ברמה הראויה הדרושה עבור ספר פרויקט בסדר הגודל הזה, שכן השתדלנו להביאו לרמה המקצועית והרצינית ביותר שיכלנו.

5.2 מסקנות

לסיכום, אנו מרוצים מהפרויקט ומספר הפרויקט שלנו, הצלחנו להביא את מה שרצינו לידי ביצוע.

הצלחנו לעמוד בהצעת הפרויקט בלי לסטות מדי מהמטרה הסופית. בתחילת הפרויקט "נשרף" זמן רב בלהמתין להגעת הריבים מחו"ל, דבר זה מנע מאיתנו להתקדם בפרויקט משום שהיינו צריכים את הרכיבים על מנת לראות האם הקוד עובד או לא. במהלך הפרויקט היו מקרים בהם נעצרנו אשר גזלו זמן רב ולא ידענו כיצד לפתור את הבעיה באותו העת ונעצרנו רבות במנחה שלו ובחברינו לכיתה למציאת פתרונותיה. אם היה ביכולתנו להתחיל את העבודה והתכנון על הפרויקט מהתחלה, היינו משנים מספר דברים:

היינו מקדיש יותר זמן למכניקה אם היינו יכולים על מנת להגיע לגודל ולצורה האידיאליים אליהם שאפנו להגיע במהלך הפרויקט, על מנת ליישם את הרעיון הסופי שהיה לנו בראש, מה שהביא אותנו לבחור דווקא בפרויקט זה.

היינו דואגים להזמין מספר כפול של רכיבים על מנת לא לדאוג אם הרכיב ייהרס או יישרף(כמו שקרה מס' פעמים, ובעקבות כך נאלצנו לחכות להגעת הרכיבים החדשים) ועל מנת שנוכל להשוות את התוצאות המתקבלות מן הרכיב עם רכיב אחר במקרה ואנו מקבלים תוצאות שאנו חושבים שהם לא נכונות או לא מדויקות.

פרק 6 – ביבליוגרפיה

השראת הפרויקט:

- https://www.youtube.com/watch?v=_6QIutZfsFs&t=315s

מידע על רכיב GSM:

- <https://lastminuteengineers.com/sim800l-gsm-module-arduino-tutorial/>

דפי יצרן:

- <https://www.farnell.com/datasheets/1682209.pdf>
- <http://roboromania.ro/datasheet/Arduino-Mega-2560-roboromania.pdf>
- https://img.filipeflop.com/files/download/Datasheet_SIM800L.pdf
- <http://www.ti.com/lit/ds/symlink/ina219.pdf>
- <http://bienonline.magix.net/public/projekte-teedipper/SG90%20g%20Micro%20Servo.pdf>
- <https://www.orientdisplay.com/pdf/CharFull/AMC1602AI2C-Full.pdf>

פרק 7 - נספחים

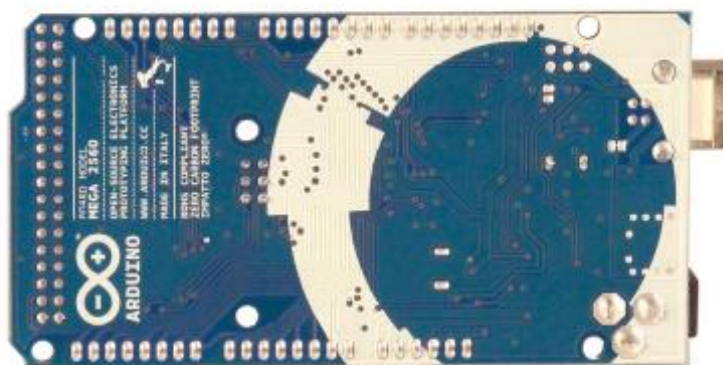
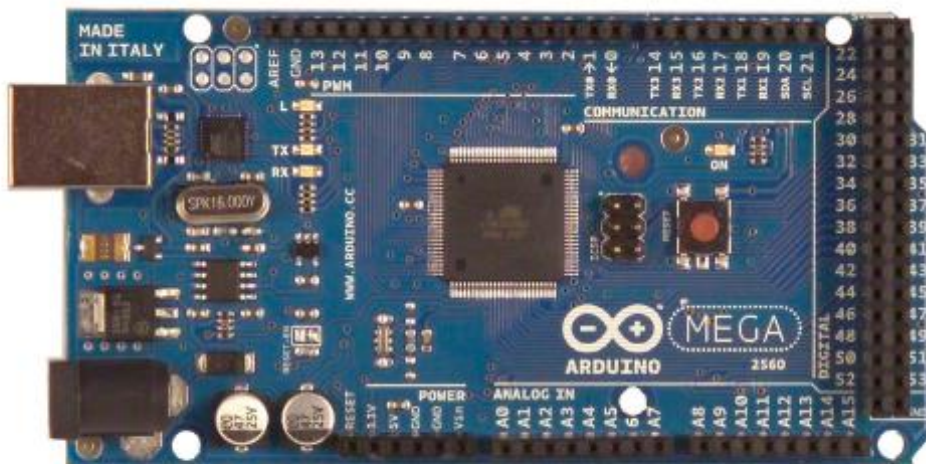
7.1 רשימת רכיבים

כמות	רכיב
1	ארדואינו אונו
1	ארדואינו מגה
1	מסך 16x2 LCD
1	מדידת מתח/זרם INA219
4	נגדי אור LDR
2	מנוע סרבו SG90
1	לוח סולארי
4	נגד $1k\Omega$
1	רכיב תקשורת GSM SIM800L EVB

7.2 דפי יצרן

7.2.1 ארדואינו מגה:

Arduino Mega 2560



Overview

The Arduino Mega 2560 is a microcontroller board based on the ATmega2560 ([datasheet](#)). It has 54 digital input/output pins (of which 14 can be used as PWM outputs), 16 analog inputs, 4 UARTs (hardware serial ports), a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. The Mega is compatible with most shields designed for the Arduino Duemilanove or Diecimila.

Schematic & Reference Design

EAGLE files: [arduino-mega2560-reference-design.zip](#)

Schematic: [arduino-mega2560-schematic.pdf](#)

Summary

Microcontroller	ATmega2560
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	54 (of which 14 provide PWM output)
Analog Input Pins	16
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	256 KB of which 8 KB used by bootloader
SRAM	8 KB
EEPROM	4 KB
Clock Speed	16 MHz

Power

The Arduino Mega can be powered via the USB connection or with an external power supply. The power source is selected automatically.

External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the Gnd and Vin pin headers of the POWER connector.

The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.

The Mega2560 differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega8U2 programmed as a USB-to-serial converter.

The power pins are as follows:

- + VIN. The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.
- + 5V. The regulated power supply used to power the microcontroller and other components on the board. This can come either from VIN via an on-board regulator, or be supplied by USB or another regulated 5V supply.
- + 3V3. A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
- + GND. Ground pins.

Memory

The ATmega2560 has 256 KB of flash memory for storing code (of which 8 KB is used for the bootloader), 8 KB of SRAM and 4 KB of EEPROM (which can be read and written with the [EEPROM library](#)).

Input and Output

Each of the 54 digital pins on the Mega can be used as an input or output, using [pinMode\(\)](#), [digitalWrite\(\)](#), and [digitalRead\(\)](#) functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms. In addition, some pins have specialized functions:

- + Serial: 0 (RX) and 1 (TX); Serial 1: 19 (RX) and 18 (TX); Serial 2: 17 (RX) and 16 (TX); Serial 3: 15 (RX) and 14 (TX). Used to receive (RX) and transmit (TX) TTL serial data. Pins 0 and 1 are also connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip.
- + External Interrupts: 2 (interrupt 0), 3 (interrupt 1), 18 (interrupt 5), 19 (interrupt 4), 20 (interrupt 3), and 21 (interrupt 2). These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the [attachInterrupt\(\)](#) function for details.
- + PWM: 0 to 13. Provide 8-bit PWM output with the [analogWrite\(\)](#) function.
- + SPI: 50 (MISO), 51 (MOSI), 52 (SCK), 53 (SS). These pins support SPI communication using the [SPI library](#). The SPI pins are also broken out on the ICSP header, which is physically compatible with the Uno, Duemilanove and Diecimila.
- + LED: 13. There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.
- + I²C: 20 (SDA) and 21 (SCL). Support I²C (TWI) communication using the [Wire library](#) (documentation on the Wiring website). Note that these pins are not in the same location as the I²C pins on the Duemilanove or Diecimila.

The Mega2560 has 16 analog inputs, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though is it possible to change the upper end of their range using the AREF pin and [analogReference\(\)](#) function.

There are a couple of other pins on the board:

- + AREF. Reference voltage for the analog inputs. Used with [analogReference\(\)](#).
- + Reset. Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

Communication

The Arduino Mega2560 has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers. The ATmega2560 provides four hardware UARTs for TTL (5V) serial communication. An ATmega8U2 on the board channels one of these over USB and provides a virtual com port to software on the computer (Windows machines will need a .inf file, but OSX and Linux machines will recognize the board as a COM port automatically. The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the board. The RX and TX LEDs on the board will flash when data is being transmitted via the ATmega8U2 chip and USB connection to the computer (but not for serial communication on pins 0 and 1).

A [SoftwareSerial library](#) allows for serial communication on any of the Mega2560's digital pins.

The ATmega2560 also supports I2C (TWI) and SPI communication. The Arduino software includes a Wire library to simplify use of the I2C bus; see the [documentation on the Wiring website](#) for details. For SPI communication, use the [SPI library](#).

Programming

The Arduino Mega can be programmed with the Arduino software ([download](#)). For details, see the [reference](#) and [tutorials](#).

The ATmega2560 on the Arduino Mega comes preburned with a [bootloader](#) that allows you to upload new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol ([reference](#), [C header files](#)).

You can also bypass the bootloader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header; see [these instructions](#) for details.

The ATmega8U2 firmware source code is available [in the Arduino repository](#). The ATmega8U2 is loaded with a DFU bootloader, which can be activated by connecting the solder jumper on the back of the board (near the map of Italy) and then resetting the 8U2. You can then use [Atmel's FLIP software](#) (Windows) or the [DFU programmer](#) (Mac OS X and Linux) to load a new firmware. Or you can use the ISP header with an external programmer (overwriting the DFU bootloader). See [this user-contributed tutorial](#) for more information.

7.2.2 ארדואינו אונו:

Input Voltage (limits)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
Analog Input Pins	6
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328) of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328)
EEPROM	1 KB (ATmega328)
Clock Speed	16 MHz

Schematic & Reference Design

EAGLE files: [arduino-uno-Rev3-reference-design.zip](#) (NOTE: works with Eagle 6.0 and newer)
Schematic: [arduino-uno-Rev3-schematic.pdf](#)

Note: The Arduino reference design can use an Atmega8, 168, or 328. Current models use an ATmega328, but an Atmega8 is shown in the schematic for reference. The pin configuration is identical on all three processors.

Power

The Arduino Uno can be powered via the USB connection or with an external power supply. The power source is selected automatically.

External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the Gnd and Vin pin headers of the POWER connector.

The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.

The power pins are as follows:

- **VIN.** The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.
- **5V.** This pin outputs a regulated 5V from the regulator on the board. The board can be supplied with power either from the DC power jack (7 - 12V), the USB connector (5V), or the VIN pin of the board (7-12V). Supplying voltage via the 5V or 3.3V pins bypasses the regulator, and can damage your board. We don't advise it.
- **3V3.** A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
- **GND.** Ground pins.

Memory

The ATmega328 has 32 KB (with 0.5 KB used for the bootloader). It also has 2 KB of SRAM and 1 KB of EEPROM (which can be read and written with the [EEPROM library](#)).

Input and Output

Each of the 14 digital pins on the Uno can be used as an input or output, using [pinMode\(\)](#), [digitalWrite\(\)](#), and [digitalRead\(\)](#) functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms. In addition, some pins have specialized functions:

- **Serial: 0 (RX) and 1 (TX).** Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip.
- **External Interrupts: 2 and 3.** These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the [attachInterrupt\(\)](#) function for details.
- **PWM: 3, 5, 6, 9, 10, and 11.** Provide 8-bit PWM output with the [analogWrite\(\)](#) function.

- **SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK).** These pins support SPI communication using the [SPI library](#).
- **LED: 13.** There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.

The Uno has 6 analog inputs, labeled A0 through A5, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though it is possible to change the upper end of their range using the AREF pin and the [analogReference\(\)](#) function. Additionally, some pins have specialized functionality:

- **TWI: A4 or SDA pin and A5 or SCL pin.** Support TWI communication using the [Wire library](#).

There are a couple of other pins on the board:

- **AREF.** Reference voltage for the analog inputs. Used with [analogReference\(\)](#).
- **Reset.** Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

See also the [mapping between Arduino pins and ATmega328 ports](#). The mapping for the ATmega8, 168, and 328 is identical.

Communication

The Arduino Uno has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers. The ATmega328 provides UART TTL (5V) serial communication, which is available on digital pins 0 (RX) and 1 (TX). An ATmega16U2 on the board channels this serial communication over USB and appears as a virtual com port to software on the computer. The '16U2 firmware uses the standard USB COM drivers, and no external driver is needed. However, [on Windows, a .inf file is required](#). The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the Arduino board. The RX and TX LEDs on the board will flash when data is being transmitted via the USB-to-serial chip and USB connection to the computer (but not for serial communication on pins 0 and 1).

A [SoftwareSerial library](#) allows for serial communication on any of the Uno's digital pins.

The ATmega328 also supports I2C (TWI) and SPI communication. The Arduino software includes a Wire library to simplify use of the I2C bus; see the [documentation](#) for details. For SPI communication, use the [SPI library](#).

Programming

The Arduino Uno can be programmed with the Arduino software ([download](#)). Select "Arduino Uno" from the **Tools > Board** menu (according to the microcontroller on your board). For details, see the [reference](#) and [tutorials](#).

The ATmega328 on the Arduino Uno comes preburned with a [bootloader](#) that allows you to upload new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol ([reference](#), [C header files](#)).

You can also bypass the bootloader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header; see [these instructions](#) for details.

The ATmega16U2 (or 8U2 in the rev1 and rev2 boards) firmware source code is available. The ATmega16U2/8U2 is loaded with a DFU bootloader, which can be activated by:

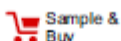
- On Rev1 boards: connecting the solder jumper on the back of the board (near the map of Italy) and then resetting the 8U2.
- On Rev2 or later boards: there is a resistor that pulling the 8U2/16U2 HWB line to ground, making it easier to put into DFU mode.

You can then use [Atmel's FLIP software](#) (Windows) or the [DFU programmer](#) (Mac OS X and Linux) to load a new firmware. Or you can use the ISP header with an external programmer (overwriting the DFU bootloader). See [this user-contributed tutorial](#) for more information.

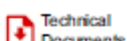
7.2.3 רכיב מתח/זרם INA219:



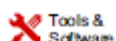
Product
Folder



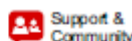
Sample &
Buy



Technical
Documents



Tools &
Software



Support &
Community



INA219

SBOS448G – AUGUST 2008 – REVISED DECEMBER 2015

INA219 Zero-Drift, Bidirectional Current/Power Monitor With I²C Interface

1 Features

- Senses Bus Voltages from 0 to 26 V
- Reports Current, Voltage, and Power
- 16 Programmable Addresses
- High Accuracy: 0.5% (Maximum) Over Temperature (INA219B)
- Filtering Options
- Calibration Registers
- SOT23-8 and SOIC-8 Packages

2 Applications

- Servers
- Telecom Equipment
- Notebook Computers
- Power Management
- Battery Chargers
- Welding Equipment
- Power Supplies
- Test Equipment

3 Description

The INA219 is a current shunt and power monitor with an I²C- or SMBUS-compatible interface. The device monitors both shunt voltage drop and bus supply voltage, with programmable conversion times and filtering. A programmable calibration value, combined with an internal multiplier, enables direct readouts of current in amperes. An additional multiplying register calculates power in watts. The I²C- or SMBUS-compatible interface features 16 programmable addresses.

The INA219 is available in two grades: A and B. The B grade version has higher accuracy and higher precision specifications.

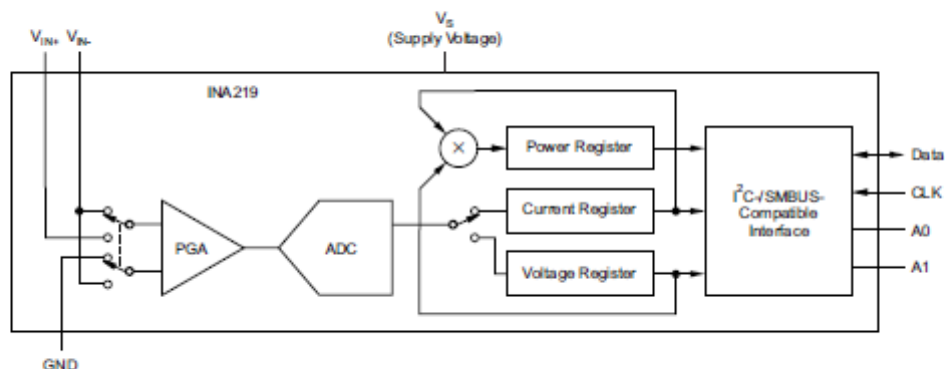
The INA219 senses across shunts on buses that can vary from 0 to 26 V. The device uses a single 3- to 5.5-V supply, drawing a maximum of 1 mA of supply current. The INA219 operates from –40°C to 125°C.

Device Information⁽¹⁾

PART NUMBER	PACKAGE	BODY SIZE (NOM)
INA219	SOIC (8)	3.91 mm × 4.90 mm
	SOT-23 (8)	1.63 mm × 2.90 mm

(1) For all available packages, see the orderable addendum at the end of the data sheet.

Simplified Schematic



6 Pin Configuration and Functions



Pin Functions

PIN			I/O	DESCRIPTION
NAME	SOT-23	SOIC		
IN+	1	8	Analog Input	Positive differential shunt voltage. Connect to positive side of shunt resistor.
IN-	2	7	Analog Input	Negative differential shunt voltage. Connect to negative side of shunt resistor. Bus voltage is measured from this pin to ground.
GND	3	6	Analog	Ground
V _S	4	5	Analog	Power supply, 3 to 5.5 V
SCL	5	4	Digital Input	Serial bus clock line
SDA	6	3	Digital I/O	Serial bus data line
A0	7	2	Digital Input	Address pin. Table 1 shows pin settings and corresponding addresses.
A1	8	1	Digital Input	Address pin. Table 1 shows pin settings and corresponding addresses.

7.1 Absolute Maximum Ratings

over operating free-air temperature range (unless otherwise noted)⁽¹⁾

		MIN	MAX	UNIT
V _S	Supply voltage		6	V
Analog Inputs IN+, IN-	Differential (V _{IN+} - V _{IN-}) ⁽²⁾	-26	26	V
	Common-mode (V _{IN+} + V _{IN-}) / 2	-0.3	26	V
SDA		GND - 0.3	6	V
SCL		GND - 0.3	V _S + 0.3	V
Input current into any pin			5	mA
Open-drain digital output current			10	mA
Operating temperature		-40	125	°C
T _J	Junction temperature		150	°C
T _{stg}	Storage temperature	-65	150	°C

(1) Stresses beyond those listed under *Absolute Maximum Ratings* may cause permanent damage to the device. These are stress ratings only, which do not imply functional operation of the device at these or any other conditions beyond those indicated under *Recommended Operating Conditions*. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

(2) V_{IN+} and V_{IN-} may have a differential voltage of -26 to 26 V; however, the voltage at these pins must not exceed the range -0.3 to 26 V.

7.2.4 מנוע סרבו SG90:

TowerPro SG90 - Micro Servo



Basic Information

Modulation: Analog

Torque: **4.8V:** 25.0 oz-in (1.80 kg-cm)

Speed: **4.8V:** 0.10 sec/60°

Weight: 0.32 oz (9.0 g)

Dimensions:

Length: 0.91 in (23.1 mm)

Width: 0.48 in (12.2 mm)

Height: 1.14 in (29.0 mm)

Motor Type: 3-pole

Gear Type: Plastic

Rotation/Support: Bushing

Additional Specifications

Rotational Range: 180°

Pulse Cycle: ca. 20 ms

Pulse Width: 500-2400 μ s

:16x2 LCD on 7.2.5

4. Absolute Maximum Ratings

Item		Symbol	Min	Max	Unit
Input Voltage		V_I	-0.3	$V_{DD}+0.3$	V
Supply Voltage For Logic		$V_{DD}-V_{SS}$	-0.3	5.5	V
Supply Voltage For LCD		$V_{DD}-V_0$	$V_{dd}-7.0$	$V_{dd}+0.3$	V
Wide Temperature LCM	Operating Temp.	T_{op}	-20	70	°C
	Storage Temp.	T_{str}	-30	80	°C

5. Electrical Characteristics

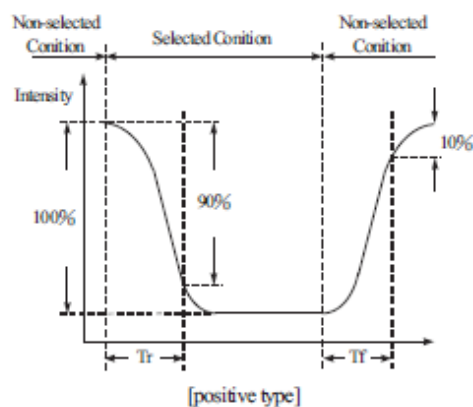
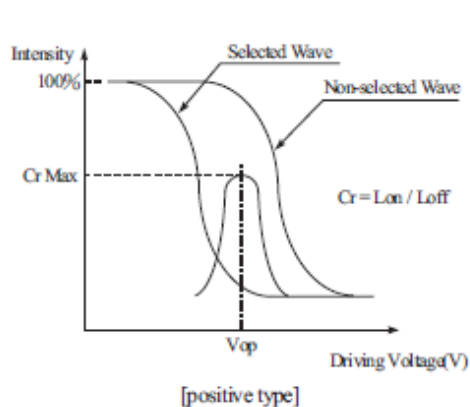
Item	Symbol	Condition	Min	Typ	Max	Unit
Supply Voltage For Logic	$V_{DD}-V_{SS}$	—	4.5	5.0	5.5	V
Supply Voltage For LCD	$V_{DD}-V_0$	$T_a=25^{\circ}\text{C}$	4.5	5.0	5.5	V
Input High Volt.	V_{IH}	—	$0.7 V_{DD}$	—	V_{DD}	V
Input Low Volt.	V_{IL}	—	V_{SS}	—	$0.3 V_{DD}$	V
Supply Current	I_{DD}	$V_{DD}=5\text{V}$	0.8	1.2	1.5	mA
Supply Voltage of Yellow-green backlight	V_{LED}	Forward current = 120 mA Number of LED die 2x12= 24	3.8	4.1	4.3	V
Supply Voltage of White backlight	V_{LED}	Forward current = 30 mA Number of LED die 1x2= 2	3.8	4.1	4.3	V

6. Optical Characteristics

Item	Symbol	Condition	Min	Typ	Max	Unit
View Angle	(V) θ	$CR \geq 2$	-20	—	35	deg
	(H) ϕ	$CR \geq 2$	-30	—	30	deg
Contrast Ratio	CR	—	—	3	—	—
Response Time	T rise	—	—	—	250	ms
	T fall	—	—	—	250	ms

Definition of Operation Voltage (V_{op})

Definition of Response Time (T_r , T_f)



Conditions:

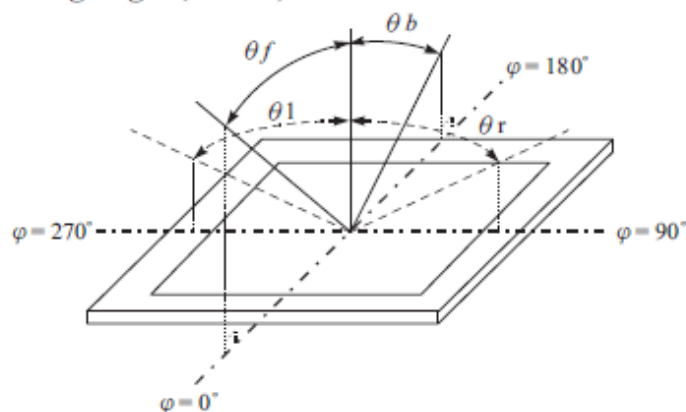
Operating Voltage: V_{op}

Viewing Angle (θ , ϕ): 0° , 0°

Frame Frequency: 64 HZ

Driving Waveform: 1/N duty, 1/a bias

Definition of viewing angle ($CR \geq 2$)



7.2.6 רכיב תקשורת GSM SIM800L EVB

1. Introduction

This document describes SIM800L hardware interface in great detail.

This document can help user to quickly understand SIM800L interface specifications, electrical and mechanical details. With the help of this document and other SIM800L application notes, user guide, users can use SIM800L to design various applications quickly.

2. SIM800L Overview

SIM800L is a quad-band GSM/GPRS module, that works on frequencies GSM850MHz, EGSM900MHz, DCS1800MHz and PCS1900MHz. SIM800L features GPRS multi-slot class 12/ class 10 (optional) and supports the GPRS coding schemes CS-1, CS-2, CS-3 and CS-4.

With a tiny configuration of 15.8*17.8*2.4mm, SIM800L can meet almost all the space requirements in user applications, such as smart phone, PDA and other mobile devices.

SIM800L has 88pin pads of LGA packaging, and provides all hardware interfaces between the module and customers' boards.

- Support 5*5*2 keypads
- One full modem serial port, user can configure two serial ports
- One USB, the USB interfaces can debug, download software
- Audio channel which includes two microphone input; a receiver output and a speaker output
- Programmable general purpose input and output.
- A SIM card interface
- Support FM
- Support one PWM

SIM800L is designed with power saving technique so that the current consumption is as low as 0.7mA in sleep mode.

2.1. SIM800L Key Features

Table 1: SIM800L key features

Feature	Implementation
Power supply	3.4V ~4.4V
Power saving	typical power consumption in sleep mode is 0.7mA (AT+CFUN=0)
Frequency bands	<ul style="list-style-type: none">● Quad-band: GSM 850, EGSM 900, DCS 1800, PCS 1900. SIM800L can search the 4 frequency bands automatically. The frequency bands can also be set by AT command "AT+CBAND". For details, please refer to document [1].● Compliant to GSM Phase 2/2+
Transmitting power	<ul style="list-style-type: none">● Class 4 (2W) at GSM 850 and EGSM 900● Class 1 (1W) at DCS 1800 and PCS 1900
GPRS connectivity	<ul style="list-style-type: none">● GPRS multi-slot class 12 (default)● GPRS multi-slot class 1~12 (option)
Temperature range	<ul style="list-style-type: none">● Normal operation: -40°C ~ +85°C

	<ul style="list-style-type: none"> Storage temperature -45°C ~ +90°C
Data GPRS	<ul style="list-style-type: none"> GPRS data downlink transfer: max. 85.6 kbps GPRS data uplink transfer: max. 85.6 kbps Coding scheme: CS-1, CS-2, CS-3 and CS-4 PAP protocol for PPP connect Integrate the TCP/IP protocol. Support Packet Broadcast Control Channel (PBCCH) CSD transmission rates: 2.4, 4.8, 9.6, 14.4 kbps
CSD	<ul style="list-style-type: none"> Support CSD transmission
USSD	<ul style="list-style-type: none"> Unstructured Supplementary Services Data (USSD) support
SMS	<ul style="list-style-type: none"> MT, MO, CB, Text and PDU mode SMS storage: SIM card
SIM interface	Support SIM card: 1.8V, 3V
External antenna	Antenna pad
Audio features	Speech codec modes: <ul style="list-style-type: none"> Half Rate (ETS 06.20) Full Rate (ETS 06.10) Enhanced Full Rate (ETS 06.50 / 06.60 / 06.80) Adaptive multi rate (AMR) Echo Cancellation Noise Suppression
Serial port and debug port	Serial port: <ul style="list-style-type: none"> Full modem interface with status and control lines, unbalanced, asynchronous. 1200bps to 115200bps. Can be used for AT commands or data stream. Support RTS/CTS hardware handshake and software ON/OFF flow control. Multiplex ability according to GSM 07.10 Multiplexer Protocol. Autobauding supports baud rate from 1200 bps to 57600bps. upgrading firmware Debug port: <ul style="list-style-type: none"> USB_DM and USB_DP Can be used for debugging and upgrading firmware.
Phonebook management	Support phonebook types: SM, FD, LD, RC, ON, MC.
SIM application toolkit	GSM 11.14 Release 99
Real time clock	Support RTC
Timing functions	Use AT command set
Physical characteristics	Size:15.8*17.8*2.4mm Weight:1.35g
Firmware upgrade	Main serial port or USB port.

2.3. Functional Diagram

The following figure shows a functional diagram of SIM800L:

- GSM baseband
- GSM RF
- Antenna interface
- Other interface

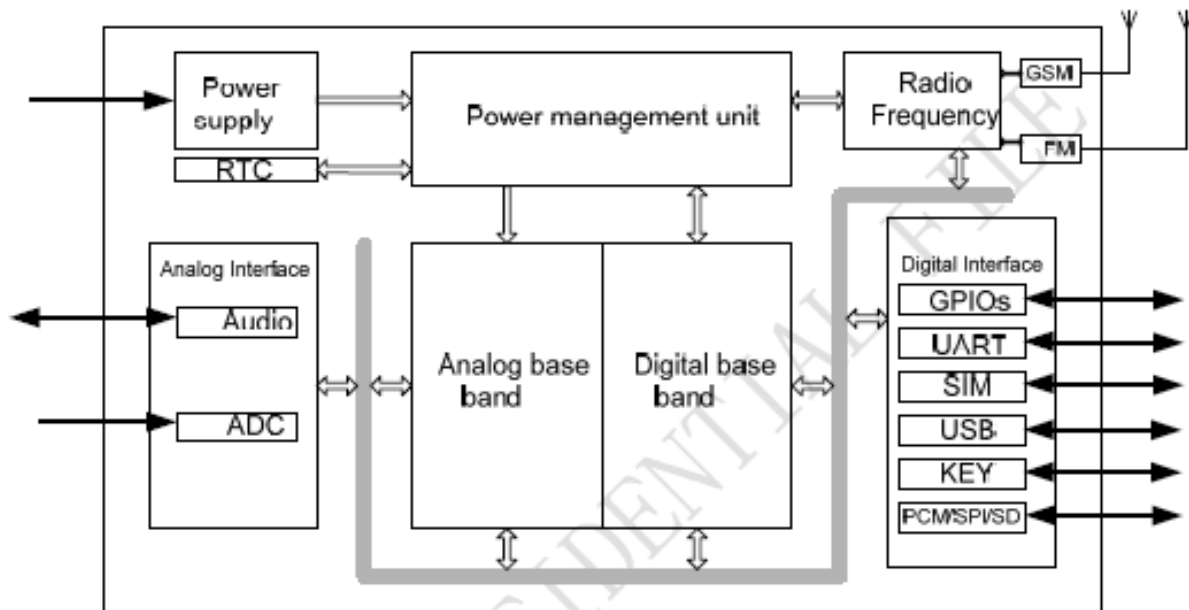


Figure 1: SIM800L functional diagram

4.1. Power Supply

The power supply range of SIM800L is from 3.4V to 4.4V. Recommended voltage is 4.0V. The transmitting burst will cause voltage drop and the power supply must be able to provide sufficient current up to 2A. For the VBAT input, a bypass capacitor (low ESR) such as a 100 μ F is strongly recommended.

Increase the 33PF and 10PF capacitors can effectively eliminate the high frequency interference. A 5.1V/500mW Zener diode is strongly recommended, the diode can prevent chip from damaging by the voltage surge. These capacitors and Zener diode should be placed as close as possible to SIM800L VBAT pins.

7.2.7 פאנל סולארי:

Number of Panels: 1

Type: Solar Panel

Foldable Solar Panel: No

Flexible Solar Panel: No

Brand Name: Overfly

Max. Power: 1W

Material: Polycrystalline Silicon

Number of Cells: 1

Nominal Capacity: 6V 1W

Size: 110*60mm

is_customized: Yes

Model Number: 6V 1W 0.16mA Solar Panel

Unit Type: piece

Package Weight: 0.03kg (0.07lb.)

Package Size: 10cm x 10cm x 10cm (3.94in x 3.94in x 3.94in)