**פרויקט סיום**

**שם הקורס והמכללה:** רובוטיקה למדעי המחשב, מכון טכנולוגי חולון (HIT).

**שם הפרויקט:** רובוט האכלה לחיות מחמד.

**שם המגישה:** לירון לוי.

**תיאור הפרויקט:** רובוט האכלת חיות מחמד - חלקי הרובוט מכילים מיכל אותו ממלאים באוכל של חיית המחמד. הרובוט מוציא 2 מנות אוכל במהלך היום בזמנים ע"פ דרישת המשתמש ובתנאי שהקערה ריקה (חיית המחמד סיימה את המנה הקודמת).

**הסבר הפעלה:** שימוש בלוח וסביבת Arduino וקוד בשפת Arduino המבוססת על שפת C++.

**הסבר על סביבת הפרויקט:**

Arduinoהוא לוח פיתוח ופלטפורמה בקוד פתוח שכולל סביבת פיתוח קלה ונוחה למשתמש.

ה-Arduino מקבל קלט מחיישנים (לדוג': חיישן קרבה, חיישן אור, כפתור) מעבד את הנתונים, ולפי התוכנה שכתובה בו מחליט האם ואיך להפעיל פלט מסוים כמו נורה, מנוע או מסך.

בליבו של ה-Arduino יושב הבקר .ATmega328 בקר זה אחראי על ביצוע כל הפקודות שאנחנו כותבים בסביבת הפיתוח ובשמירתם בזיכרון שלו.

שפת התכנות של ה-Arduino נותנת לנו גישה להפעיל ולכבות פינים מסוימים בבקר ובנוסף יש לנו גישה אל הבקר דרך הפינים השקועים שעל לוח ה-Arduino.

ניתן לראות אם כן שלוח ה-Arduino פשוט נותן לנו גישה אל בקר ה- ATmega ודרך חיבורו למחשב אנחנו יכולים לכתוב תוכנה בסביבת הפיתוח שתגיד לבקר מה לעשות.

כדי להפעיל את ה-Arduino צריך לתת לו כוח אשר יכול להיות מסופק מכניסת ה- USB של המחשב, מכניסת החשמל או מבטרייה חיצונית.

האונו מופעל ע”י כבל USB אשר מספק לו מתח של 5 וולט וגם מאפשר העברת נתונים – כתיבת התוכנה מסביבת הפיתוח שעל המחשב ל-Arduino.

שפת התכנות שבה אנו משתמשים היא שפת Arduino ומבוססת על שפת C++.

ישנן שתי פונקציות שחייבות להופיע בכל תוכנית Arduino שאנחנו כותבים:

void setupהיא פונקציה שרצה פעם אחת כשה-Arduino מופעל. Void loop היא פונקציה שרצה בצורה אינסופית מרגע שה-Arduino נדלק ועד שאנחנו מכבים אותו.

**פירוט שלבי הפיתוח:**

1. הבנת לוח ה-Arduino ואיך עובדים איתו.
2. הורדת סביבת קוד ל-Arduino.
3. התחלת כתיבת הקוד. בכתיבה לסביבת Arduino הקוד מתחלק ל2 מתודות:

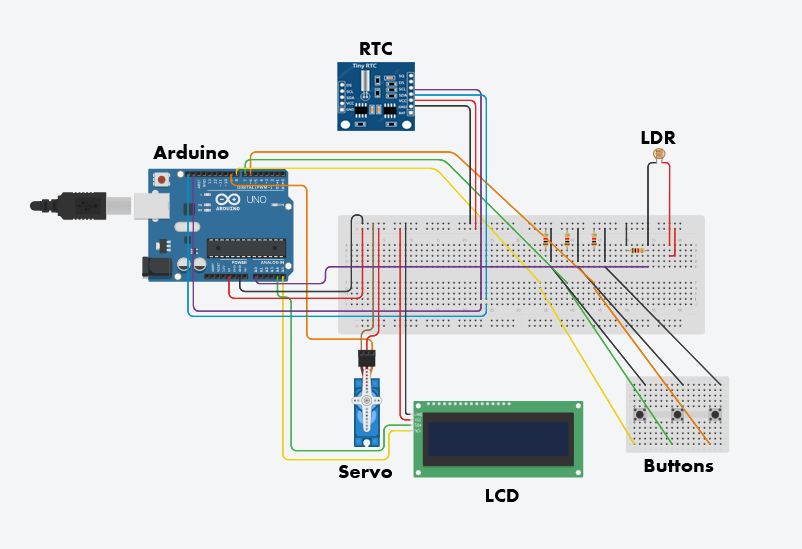
void setup(), void loop().

1. קידוד מערכת השעון וחיבור הRTC ds1307-.

הורדת הספריות RTClib.h, LiquideCrystal\_12C.h וקריאה אליהם מהקוד. ה-RTC שומר את השעה הנוכחית ושעת ההאכלה (ללא דקות ההאכלה) שהמשתמש הכניס במידה והמכשיר נותק מהחשמל. ברגע שהמכשיר יחובר בחזרה לחשמל, הצג יראה את השעה הנוכחית.

1. יצירת מתודות על מנת לקבוע את השעה הנוכחית ושעות ההאכלה, בנוסף ישנה מתודה בשם StoreAgg שמאחסנת ב-RTC את השעות שנקבעו ע"י המשתמש.
2. הלחמנו פינים לצג וחיברנו אותו כדי שיציג את השעה הנוכחית שהמשתמש בחר ואת שעת ההאכלה שהמשתמש בחר.
3. קריאה למתודה DisplayDateTime המציגה את למסך את השעה הנוכחית ושעת ההאכלה.
4. חיברנו את המנוע micro servo ללוח Arduino ולדסקית שפותחת וסוגרת את מיכל האוכל.
5. חיברנו למטריצה 3 כפתורים ונגדים בהתאמה בכדי לשמור על מעגל חשמלי תקין.
6. חיברנו למיכל אוכל משטח אלכסוני המחבר בין מיכל האוכל לקערת האוכל.
7. בסוף המשטח מחובר חיישן אור LDR העוזר לדעת האם יש אוכל בקערה או שהקערה ריקה (על פי החזרי אור).
8. הכנסת כל האלקטרוניקה לקופסת פלסטיק, חיתוכה להצגת המסך, סידור הכפתורים, יצירת מקומות ליציאות החשמל וחיבורי המנוע ה-Servo וחיישן האור LDR.

**סימולציית החיבורים:**



**מסקנות והמלצות להמשך הפיתוח:**

מסקנות:

• סוג הנגד משפיע מאוד על תוצאות החיישן, נתקלנו בקשיים מסוימים עם חיישן האור שאותם פתרנו בעזרת החלפת הנגד מ220Ω ל-10k.

• חיבורי ה-Arduino מאוד רגישים ושבירים, נתקלנו בתקלות מסוימות שניסינו לפתור במשך שעות ובסוף התברר שהתנתק/נשבר כבל מסוים.

• שיפוע המשטח המחבר בין המיכל לקערה משפיע על מהירות והורדת האוכל.

• לחשוב על דרכי פיתוח הפרויקט וחלקיו לפני הקמתו – עשיית רשימה מאורגנת ומסודרת של הרכיבים והחלקים הנדרשים.

• זמן השהיה בקוד (delay) בין לחיצה על כל כפתור משפיע על הקפיצות בלחיצות שלו. לחיצה ארוכה מקפיצה כמה ערכים.

• צריך להתאים את ערכי התנאי של ה-LDR לכל סביבה בה בוחרים למקם את הרובוט.

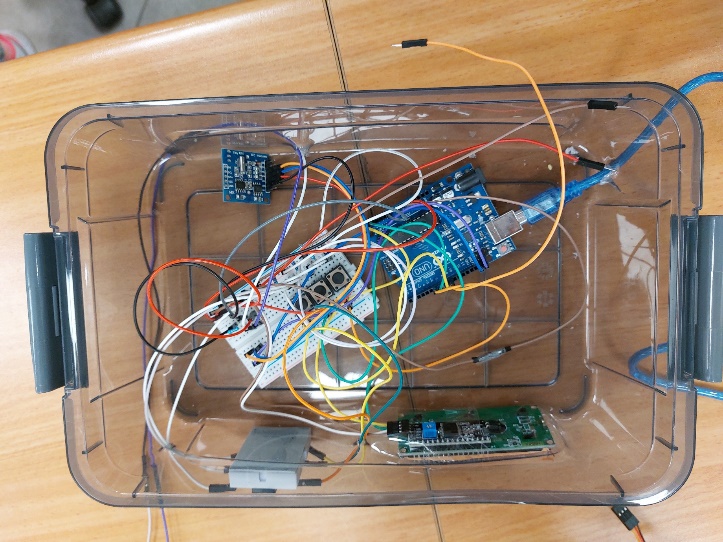
המלצות להמשך:

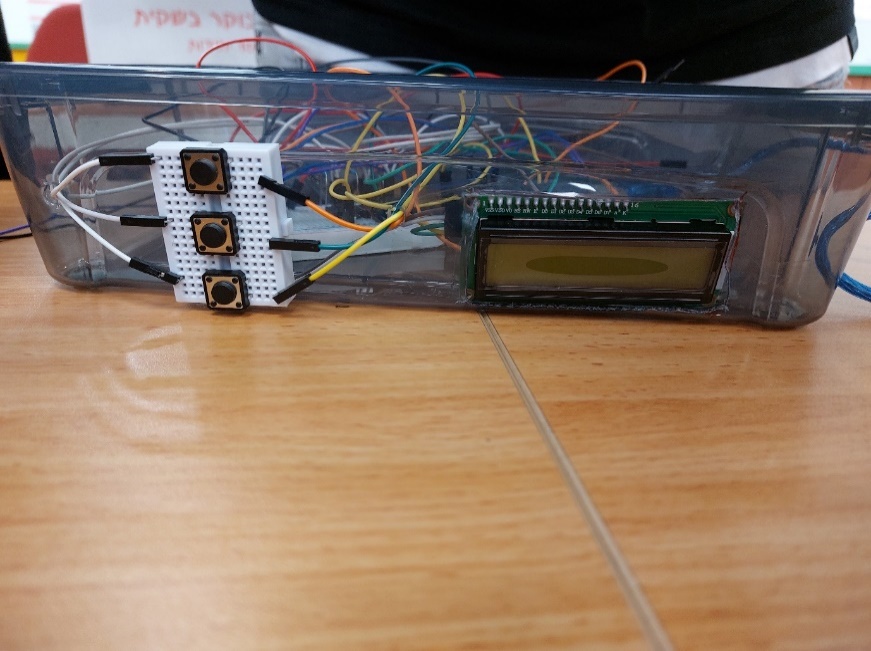
• החלפת את חיישן האור LDR בחיישן משקל כדי לדעת את משקל המנה המדויק וכך להתאים את כמות האוכל לחיית המחמד.

• בחירת חיישן משקל שיהיה מותאם למשקלים קטנים.

• החלפת המיקרו servo 180 למיקרו servo 360 כך שתתקבל זווית פתיחה גדולה יותר ובכך לאפשר לאוכל לצאת בצורה חלקה יותר.

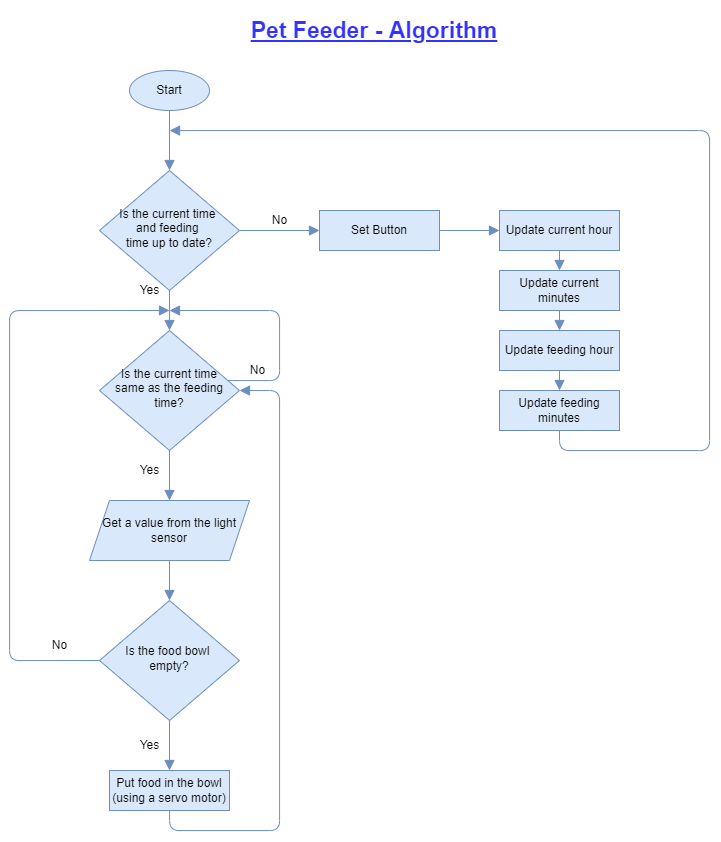
**צילומי הפרויקט:**







**תרשים זרימה של האלגוריתם:**

****

**קישורים למידע על רכיבי הפרויקט:**

• Arduino Uno: <https://docs.arduino.cc/hardware/uno-rev3> .

• Servo: <http://www.ee.ic.ac.uk/pcheung/teaching/DE1_EE/stores/sg90_datasheet.pdf>

• RTC: <https://www.openhacks.com/uploadsproductos/tiny_rtc_-_elecrow.pdf>

• Lcd: <http://wiki.sunfounder.cc/index.php?title=I2C_LCD2004>

• LDR: <https://startingelectronics.org/beginners/arduino-tutorial-for-beginners/arduino-LDR-tutorial/>