



אוניברסיטת בן-גוריון בנגב

Ben-Gurion University of the Negev

בית הספר להנדסת חשמל ומחשבים

דו"ח מסכם לפרויקט גמר

בקורס: "מבנה מחשבים ספרתיים" – 1-361-4191

Radar Detector System

תכנון ומימוש מערכת רדאר לניטור וגילוי אובייקטים במרחב

תאריך הגשה: 17/08/2021

שם מדריך אחראי: יאיר

שם מגיש: אופיר תנעמי

שם מגיש: אריאל משה

ת"ז: 204541528

ת"ז: 312549660

תוכן עניינים

3.....	הגדרות ומטרת הפרויקט
5.....	ביצועי החומרה והתוכנה
6.....	תרשים זרימה של ארכיטקטורת התוכנה
7.....	ביצועים לעומת המפרט הטכני
7.....	הקצאת חיבורים
7.....	תרשים סכמתי של המעגל האלקטרוני
8.....	ממשק משתמש
9.....	מסקנות והצעות לשיפורים

הגדרות ומטרת הפרויקט

- i. תכנון ומימוש מערכת רדאר מבוססת MCU לניטור וגילוי אובייקטים במרחב באמצעות מד מרחק אולטראסוני ומנוע Servo. סריקת המרחב תבוצע בגזרה של 180 מעלות באמצעות תנועת מנוע Servo, טווח המדידה באמצעות מד המרחק נע בין 2M-4.5M. בקרת התנועה הזוויתית של מנוע Servo תהא מבוססת PWM.
- ii. במסגרת הפרויקט יפותח קוד בשפת C++/C למימוש מערכת זמן אמת מבוססת פסיקות (צד MCU) להפעלת הרכיבים וקריאת המידע ממד המרחק.
- iii. לצורך תצוגה וממשק למשתמש, ישמש מחשב PC עליו תוצג תמונת הרדאר. ה-MCU יחובר למחשב ה-PC באמצעות תקשורת טורית אסינכרונית בסטנדרט RS-232.
- iv. ממשק למשתמש בצד ה-PC יאפשר קביעת פרמטרים, שליחת קבצים ופקודות High-level ל-MCU ותצוגת תמונת הרדאר ב-PC. הממשק בצד ה-PC יכתב בשפה עילית (לבחירתכם: MATLAB, Python, C++, JAVA) או שימוש במעטפת C# - מומלץ) ויתמוך במימוש של תקשורת טורית בין הבקר ל-PC.
- v. הממשק יאפשר העברת קבצים הכוללים פקודות High-level מקודדות למימוש בצד הבקר. הקבצים בבקר יישמרו בזיכרון FLASH.

תיאור הפרויקט:

בפרויקט זה נוכל לנווט בין מספר אופציות תפריט שונות באמצעות ממשק גרפי (GUI), האופציות הקיימות אותם ממשנו:

1. מימוש מערכת רדאר (Radar Detector System) – כאשר נבחר באופציה זו תוצג סריקה דינאמית 180 מעלות סביב הרדאר ותצייר את מיקומים האובייקטים השונים בהתאם למרחקם מהרדאר.
2. טלמטר (Telemeter) – באפשרות זו המשתמש בוחר זווית קבועה, ועל המסך יוצג באופן דינאמי המרחק של האובייקט הקרוב לרדאר ברזולוציה של cm.
3. סקריפט מוד (Script Mode) – הפעלת כל המערכת בהתאם לקובץ סקריפט המכיל פקודות המוגדרות מראש, לדוגמה הפקודות המוגדרות באופציות תפריט 1 ו-2, הדלקת לדים ונורת RGB.

תיאור פונקציות:

קובץ	פונקציה	פעילות
Main.c	<i>int receive_int();</i>	מקבל int מה-PC
	<i>void servo_deg(int deg)</i>	מודד מרחק בזווית הנבחרת
	<i>void servo_scan(int left,int right)</i>	מבצע סריקה בין זווית שמאלית לימנית
	<i>Script_mode()</i>	כותב את הסקריפט לזיכרון פלאש ומריץ את הסקריפט
menuFunction.h	<i>int str2int(volatile char * str)</i>	ממיר מחרוזת לinteger
	<i>int int2str(char * str,int num)</i>	ממיר integer למחרוזת
	<i>int receive_int()</i>	מקבל int מה-PC
	<i>void receive_string(int * data)</i>	מקבל מחרוזת מה-PC
	<i>void send_ack(int data)</i>	שולח ביט אישור ל-PC
	<i>void send_ss_data(int deg,int distance)</i>	שולח את המרחק והזווית מהחיישן ל-PC
	<i>void write_seg (char * flash_ptr,int offset)</i>	כותב char לזיכרון הפלאש
	<i>char read_char(char address)</i>	קורא char מזיכרון הפלאש
	<i>int read_mem(int offset)</i>	קורא מספר char רצופים מזיכרון הפלאש
	<i>void blink_rgb(int delay,int times)</i>	מדליק את נורת ה-RGB בצבעים שונים בקצב של הדיילי
	<i>void rlc_leds(int delay,int times)</i>	מדליק את הledים ומבצע הזזה שמאלה בקצב של הדיילי
	<i>void rrc_leds(int delay,int times)</i>	מדליק את הledים ומבצע הזזה ימנית בקצב של הדיילי
	<i>void servo_deg(int deg)</i>	מודד מרחק בזווית הנבחרת
	<i>void servo_scan(int left,int right)</i>	מבצע סריקה בין זווית שמאלית לימנית
	<i>void InitGPIO(void)</i>	מקנפג את ההפורטים של הבקר בתחילת הריצה
	<i>void StopTimers()</i>	עוצר את כלל השעונים
bspGPIO.c	<i>void Timer0_A_delay_ms(int ten_mili_sec)</i>	מפעיל דיילי עם שעון TA0
	<i>void Timer1_A_delay_ms(int ten_mili_sec)</i>	מפעיל דיילי עם שעון TA1
	<i>void SS_Echo_config()</i>	מודד את המרחק מהחיישן ומחזיר בס"מ
halTimers.c		

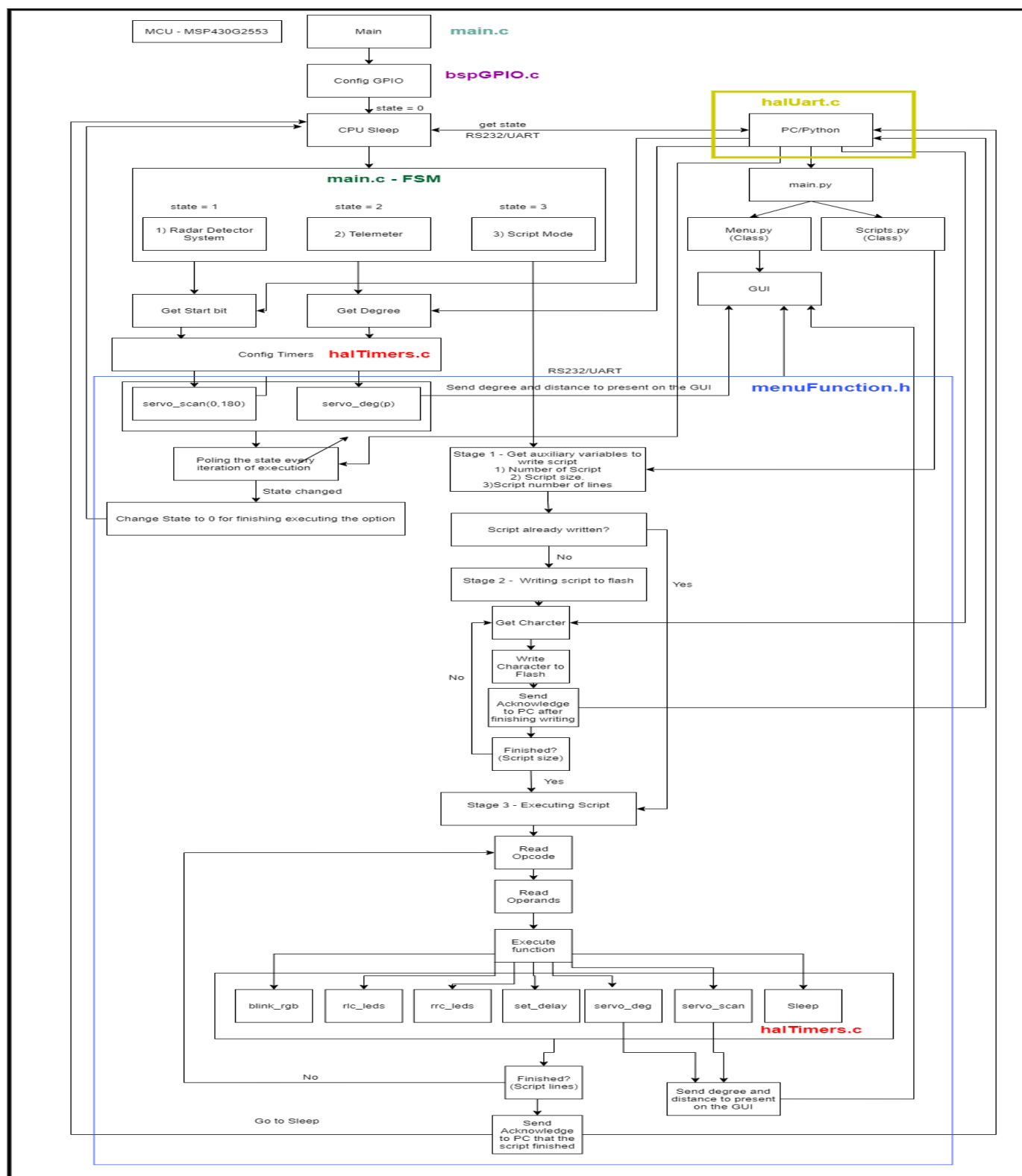
מזיז את מנוע ע"פ הזווית הרצויה	<i>void PWM_Servo_config(int deg)</i>	
יוצר פולס אתחול לתחילת פעולת החיישן	<i>int SS_Trig_config()</i>	
פסיקה של TA0.0 לצורך השהייה	<i>_interrupt void Timer0_A0(void)</i>	
פסיקה של TA0.0 לצורך השהייה	<i>_interrupt void Timer1_A0(void)</i>	
פסיקה לצורך input capture של שעות TA1.1	<i>_interrupt void Timer1_A1(void)</i>	
אתחול הרגיסטרים של מודול UART-ה	<i>void uart_config()</i>	<i>halUart.c</i>
פסיקת הקבלה RX של מודול UART-ה	<i>_interrupt void USCI0RX_ISR(void)</i>	
פסיקת השליחה TX של מודול UART-ה	<i>_interrupt void USCI0TX_ISR(void)</i>	

ביצועי החומרה והתוכנה

בצד של התוכנה השתמשנו בשפת התכנות פיתון על מנת לכתוב ממשק גרפי, ניווט בין תפריטים שונים והצגת התוצאות באופן גרפי ודינאמי, ועל מנת לבצע תקשורת בין המחשב האישי לבקר (שני השימושים נעשו ע"י שימוש במודולים קיימים בשפת התכנות). את תכנות הבקר כתבנו בשפה התכנות C על מנת לקנפג את הפורטים השונים וליצור מערכת לוגיקה אותה יבצע הבקר כפי שמפורט בתרשים. מבחינת החומרה השתמשנו בבקר $MSP430$, ופרוטוקול תקשורת $RS - 232$ (עליו הסברנו הרחבה במעבדה 4) שהיה אחראי על העברת התקשורת בין המחשב האישי לבקר. רכיבי חומרה הנוספים היו חיישן האולטראסוניק ומנוע הסרבו. המנוע היה אחראי להזזת החיישן כך שנוכל לסרוק את הסביבה 180 מעלות. באמצעות חיישן האולטראסוניק מדדנו את מרחק האובייקט הקרוב אליו. את שני הרכיבים האחרונים (מנוע הסרבו וחיישן האולטראסוניק) תפעלנו באמצעות שעוני הבקר כך שנקבל מהם את הקלט הנדרש.

תרשים זרימה של ארכיטקטורת התוכנה

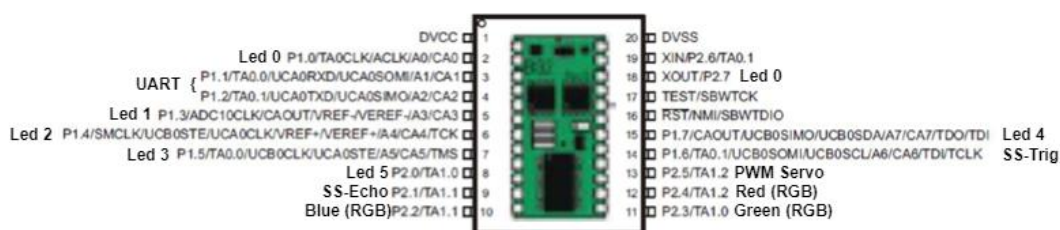
להלן תרשים זרימה של תכנון ארכיטקטורת התוכנה למימוש המערכת *Embedded*.
מחולקת לשכבות המבוססת *FSM*.



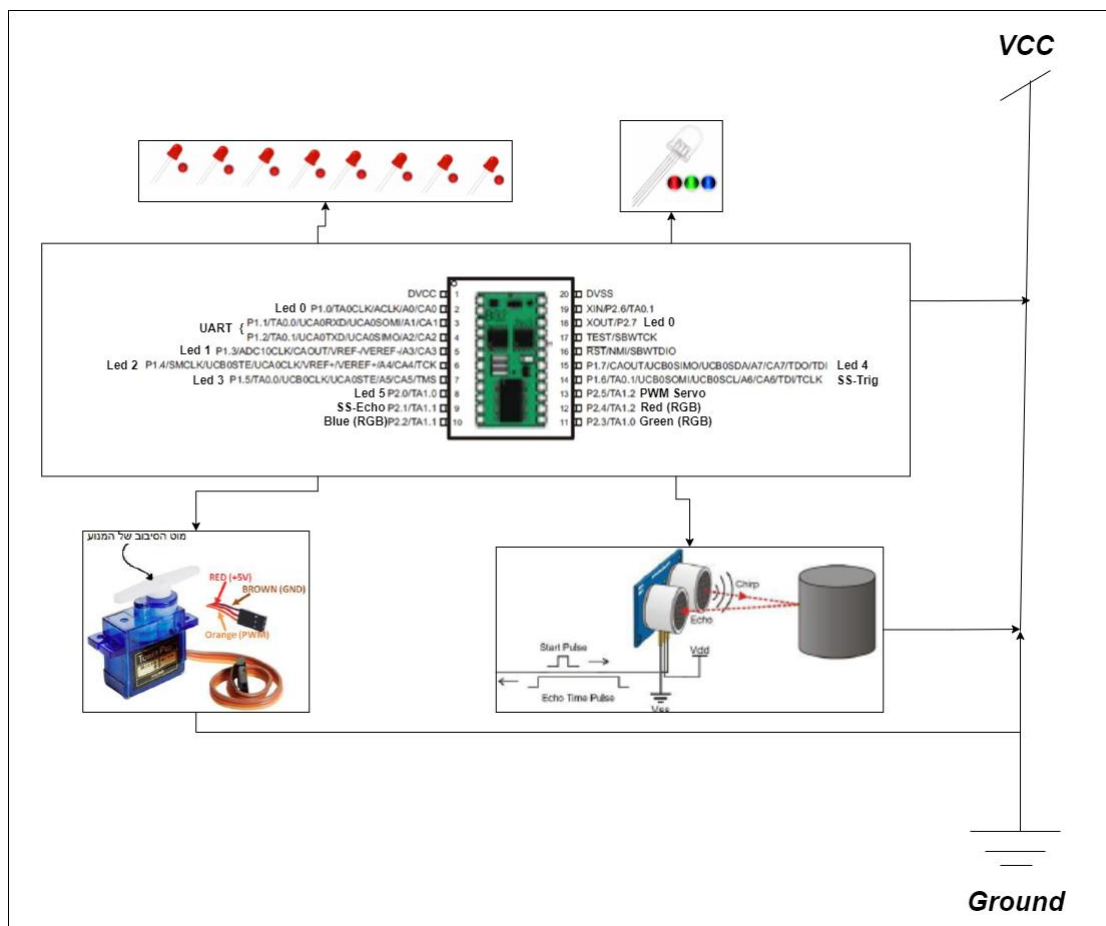
ביצועים לעומת המפרט הטכני

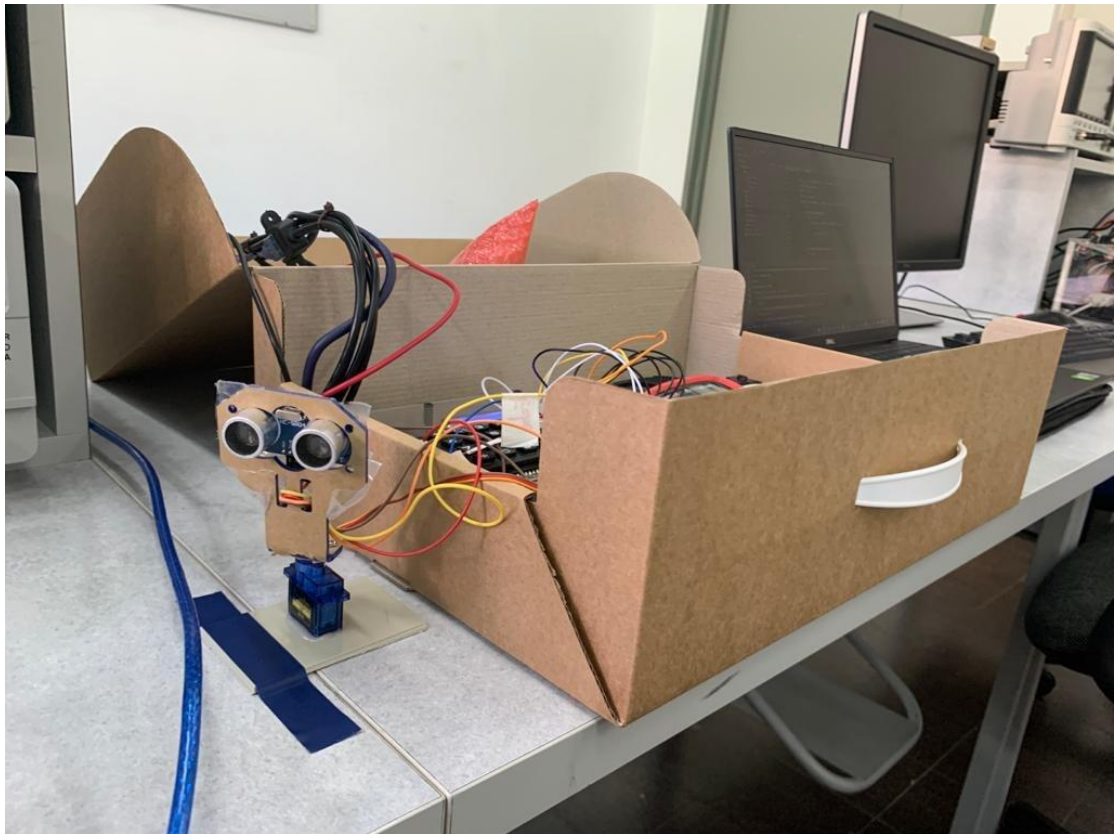
בפועל עבדנו עם קצב שעון נמוך יותר מ- $40[ms]$ מכיוון שקצב זה הכניס הרבה רעש וחוסר יציבות בדגימות. בנוסף מכיוון שאנחנו עבדנו עם מספר שעונים קטן נאלצו להשתמש באותם השעונים במשימות שונות, לכן נאלצו לבצע השהיות רבות בין פעולות שונות, על מנת לא לפגוע במהלך תקין ולגרום לחפיפה בין פעולות שלוקחות מעט זמן לפעולות שעשויות לקחת יותר זמן וכך "להידרס" ע"י שינוי פעולת השעון.

הקצאת חיבורים

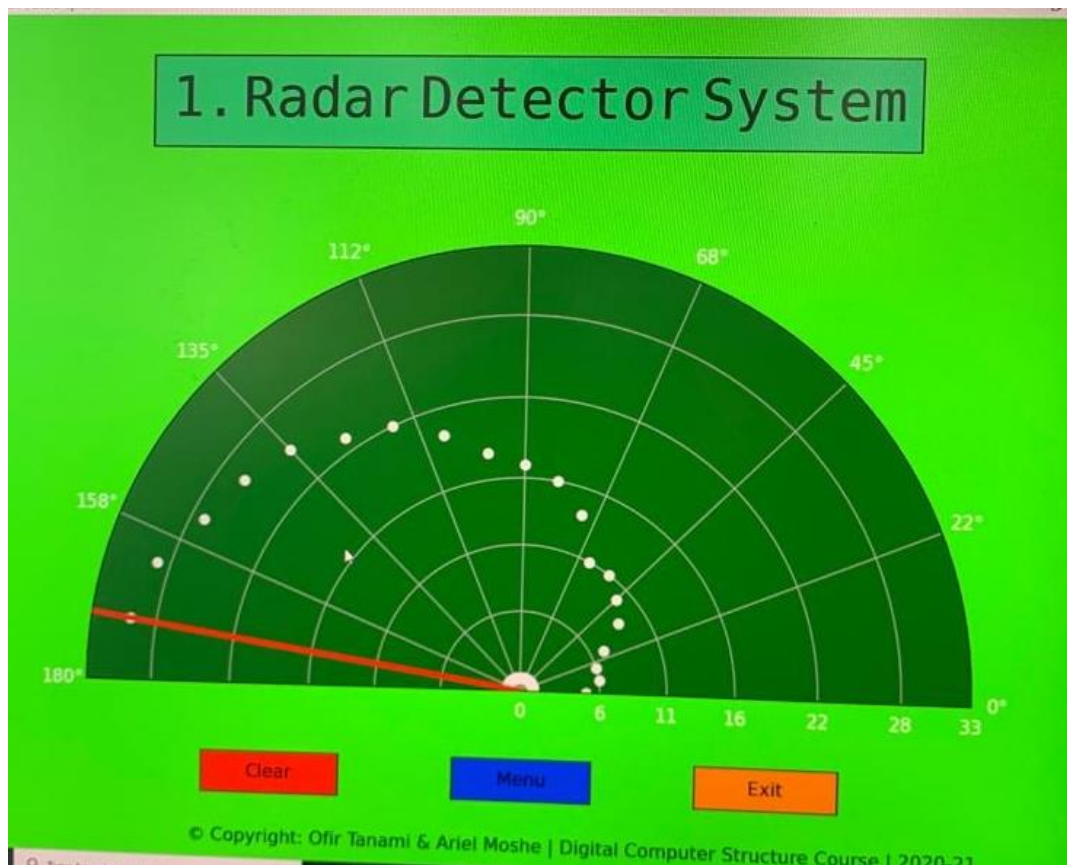


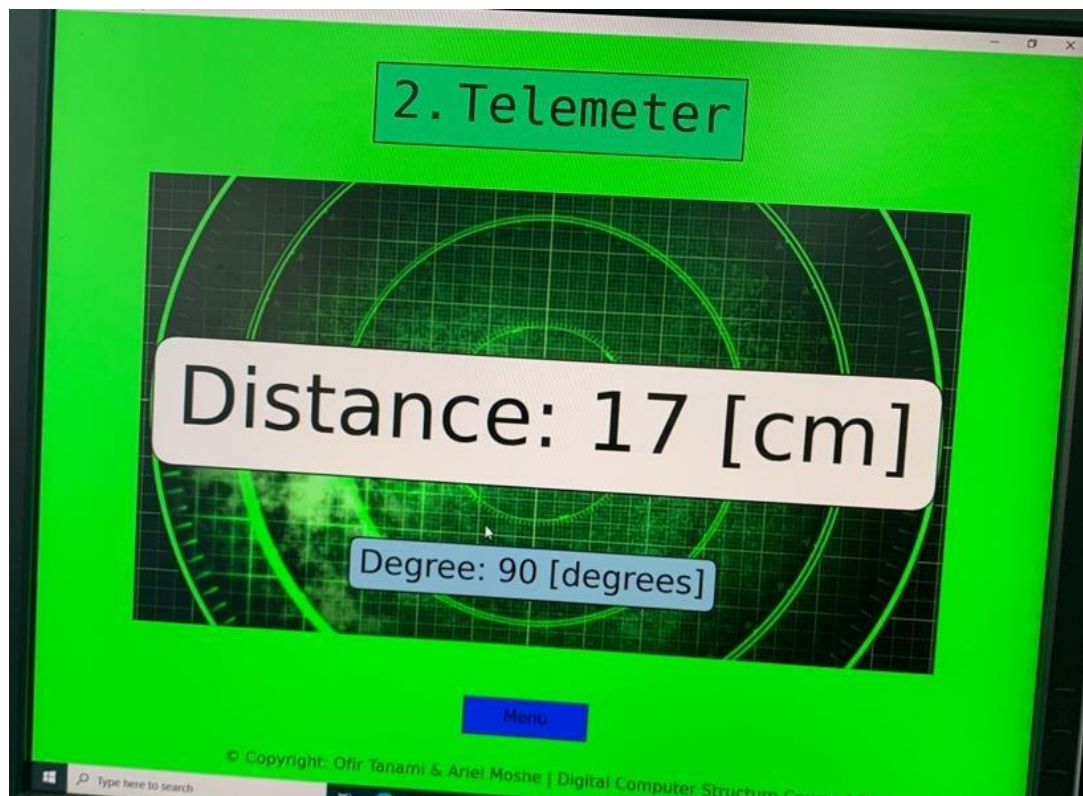
תרשים סכמתי של המעגל האלקטרוני





ממשק משתמש





מסקנות והצעות לשיפורים

- את כל החישובים היינו צריכים לעשות בצד המחשב ולא בצד הבקר, וככה לעבוד בצורה יותר יעילה, כיוון שיכולות החישוב של הבקר מוגבלות ואיטיות יותר, כגון חוסר בחומרה של מכפל לצורך פעולת כפל.
- הדגימות היו צריכות לעבור סינון על מנת להוריד רעשים. על מנת להציג מידע מדויק יותר היה ניתן להתמקד בשיפור הרזולוציה של המרחק המוצג ע"י מניעת השינוי שלו בהתאם לדגימות קודמות.
- שליחה של המידע הנמדד נשלח כמחרוזת. ניתן היה לשפר את הביצועים ע"י שליחת 2 בתים בסה"כ המייצגים את אוגר המונה של השעון המייצג את השעון של החיישן.
- להיות צמודים להוראות הצגת הפרויקט, להציג את הדברים לפי הפורמט המבוקש.