



אוניברסיטת בן-גוריון בנגב  
Ben-Gurion University of the Negev

# פרויקט גמר

קורס: מבנה מחשבים  
ספרתיים הנדסת מחשבים  
שנה ב

עומר לוכסמבורג עמנואל קופמן

13/05/2019

**מטרת הפרויקט:** ליישם אפליקציות העושות שימוש בבקר MKL25H4 ורכיביו (בדגש על accelerometers) ובחיישן ULTRASONIC, תוך כדי קשר עם מחשב (צד לקוח) דרך תקשורת לפי פרוטוקול UART.

**מבנה הפרויקט:** סביבת העבודה שלנו בפרויקט הסיום מתחלקת ל-2, צד הבקר, code warrior, שדרכו אנו מתכנתים את הבקר וה-ULTRASONIC. וסביבת המחשב, שם דרך Matlab אנו מנתחים את המידע שאנו קולטים מהבקר (דרך תקשורת בפרוטוקול UART) וגם מציגים אותו בצורה גרפית למשתמש, דרך ממשק גרפי אותו בנינו ב-Matlab. אותו ממשק משמש גם כתפריט למשתמש דרכו הוא יכול לבחור באחת משלושת האפשרויות, ועל גבי חלון הממשק יוצג גרף בהתאם לבחירת המשתמש.

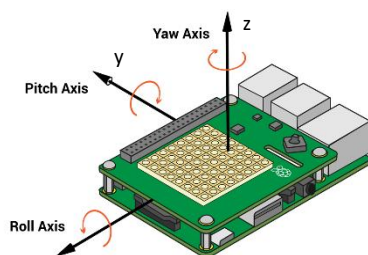
### **תיאור האפליקציות:**

1. **סריקת עומק תלת מימדית**- לאחר שהמשתמש בוחר ביישום זה, עליו לכוון עם חיישן ה-ULTRASONIC על הפינה השמאלית התחתונה של מסגרת בגודל אותו הוא יבחר כאשר במסגרת עצם דומם העומד עם מרקע אחיד מאחוריו (זאת אומרת שהמרחק מהחיישן למרקע הוא זהה בכל נקודה במסגרת, בנוסף מומלץ לסרוק עצמים אנכיים לחיישן, מעצמם מגבלות החיישן). על המשתמש לבצע מחווה עם הבקר בכך שיסיט אותו ימינה בציר y ואז הסריקה תתחיל להתבצע לאורך המרחק אותו הגדיר המשתמש. הסריקה מתבצעת כך שכל זמן שהגדרנו מראש זמן אינטרוול כך שבכל אינטרוול המחשב מבקש בקשה מהבקר שישלח את המספר המחזורים בו רגל echo של ה-ULTRASONIC נמצא על אחד לוגי (פירוט של אופן חישוב מספר המחזורים יבוא בסעיף 2). הבקר שולח דרך ה-UART את מספר המחזורים ובמאטלאב מתבצע חישוב לפי נוסחא נתונה של המרחק של החיישן מהאובייקט שנמצא מול הקולטנים של החיישן. לאחר שסריקת השורה הראשונה תתבצע, על המשתמש לבצע מחווה עם הבקר בכך שיסיטו כלפי הכיוון השלילי של ציר x. ולאחר מכן להרים את הבקר עם החיישן מעט מעלה כדי להתחיל לסרוק את השורה הבאה, וכך הלאה, כאשר הטריגר לתחילת הסריקה הוא הסטת הבקר בציר y לכיוון הסריקה. כאשר המשתמש יסיים לסרוק שורה וירצה לסיים ולראות את התוצאה הסופית עליו יהיה להסיט את הבקר בכיוון החיובי של ציר x. ולאחר סיום הסריקה יוצג למשתמש גרף המציג את סריקת העומק. אנו בונים גרף תלת מימדי כך כשאנו מחלקים את המרחב התלת מימדי לוקטור של ציר x ווקטור של ציר y ומטריצה של המרחקים שמיוצגת על ידי ציר Z (כאשר כל איבר במטריצה נמצא במיקום המתאים למיקום בו אנו סורקים ובכך מתאימים מיקום בוקטור  $x$  ו  $y$ ). כך אנו מקבלים תמונה כוללת של סריקת העומק של האובייקט.
2. **מד מרחק**- לאחר שהמשתמש בוחר ביישום זה מוצג גרף של מרחק מהחיישן כפונקציה של זמן, כאשר בכל רגע נתון הגרף מתעדכן בהתאם למרחק הנוכחי. כעת נפרט מעט על אופן

הפעולה: אנו יוצרים את PWM אשר אותה שולחים לרגל Trigger של החיישן בעזרת שעון TPM כך שרוחב הפולס שלו יהיה לפחות 10 מיקרו שניות, כך שהמרווח בין אות לאות הוא לפחות 60 מילי שניות. כאשר בסיום הפולס חיישן המרחק "יורה" גל קול לכיוון האובייקט וקולט את ההחזרים המגיעים ממנו. החיישן ממיר את החזרי גל הקול וממיר אותו לפולס היוצא מרגל Echo. את רוחב הפולס (איתו אנחנו מחשבים את המרחק מהחיישן) אנו מוצאים על ידי טיימר שני (שגם הוא מסוג TPM) בעזרת capture, אנו מוצאים את הזמן בו קרתה העלייה לאחד לוגי מרגל אקו ואת הזמן בירידה ל0 לוגי, ועל ידי חיסור הזמנים קיבלנו את רוחב הפולס. בפונקציה זו בלבד, כדי לקבל תוצאות יותר מדויקות אנו מחשבים ממוצע של 17 דגימות עוקבות של פולסים מרגל Echo ואותו שולחים בעזרת הUART למחשב בו מתבצע החישוב לפי הנוסחא שניתנה בהגדרת הפרויקט. מכיוון שהנוסחא תלויה בטמפרטורה אנו נותנים למשתמש לבחור את הטמפרטורה וכך אנו נקבל תוצאות מדויקות יותר.

בנוסף אנו מציגים את המרחק על גבי מסך הLCD שכן אנו שולחים את הטמפרטורה שנבחרה בעזרת הUART ואז עושים חישוב גם בצד הבקר של המרחק אשר אותו מציגים על המסך.

3. תזוזה דו מימדית- כאשר המשתמש בוחר במצב זה מוצג על גבי הממשק הגרפי גרף תלת מימדי של תיבה המייצגת את הבקר אשר נמצא אצל המשתמש. ועם הטיית הבקר סביב ציר x או y השרטוט התלת מימדי על המסך מוטה יחד עם הבקר לאותו כיוון. כל זה מתבצע בעזרת חיישן התאוצה הנמצא בבקר. ה-accelerometer זהו חיישן התאוצה שממנו אנו שולפים את תאוצת הבקר על ידי מדידה של שינוי הקיבול הנוצר על ידי תזוזה מכנית של המסה לשלושת הכיוונים. בתוך הבקר חיישן התאוצה מודד את התאוצה בכל כיוון, החיישן שולח לזיכרון שהגדרנו מראש את הנתונים בעזרת פרוטוקול I<sup>2</sup>C. את הנתונים האלה אנו שולחים מהבקר למחשב בעזרת הUART, בעזרת הנוסחאות:



$$\tan \phi_{roll} = \frac{G_y}{G_z} \quad \tan \theta_{pitch} = -\frac{G_x}{\sqrt{(G_y^2 + G_z^2)}}$$

x

אנו משיגים את השינוי בזווית ולפי זה אנו יודעים איך להזיז את הזוויות ה-pitch וה-roll. ובכך אנו משיגים קורלציה בין תנועת הבקר במעבדה, לבין תנועת האובייקט בגרף.

## פעולות

### בצד בקר

פעולות:

1. Void MCU\_Init(void) – פעולת המבצעת אתחול למודול ה- $I^2C$ .
2. Void Accelerometer\_Init(void) – פעולת אתחול לחיישן התנועה.
3. Void Calibrate(void) – כיול חיישן התנועה.

פסיקות:

1. Void FTM0\_IRQHandler(void) – פסיקת שעון TPM0 אשר אחראי על אות ה-PWM, שנשלח לרגל ה-trigger. פסיקה תתבצע ב-overflow.
- בפסיקה עצמה, אנו מקבלים את מספר המחזורים של שעון TPM2 שבו echo נשאר על '1' לוגי. לאחר מכן אנו מחשבים על פי הטמפרטורה, את המרחק.
2. Void FTM2\_IRQHandler(void) – פסיקת שעון TPM2 – הפועל במוד input capture, כלומר מתבצעת תפיסת שעון בעליה וירידה של הפולס היוצא מרגל ה-echo.
- בין עלית שעון וירידת שעון, אנו משתמשים במונה של השעון, לספירת מספר המחזורים.

בצד המחשב (matlab)

פעולות:

1. getAvailableComPort() – מציאת הפורט הפנוי אליו מחובר הבקר, באופן אוטומטי.
2. RangeCalculate(char) – מקבל תו בודד ('r' / 's'), ושולח אותו 17 פעמים לבקר. בפעם ה-17 ינתן ערך הממוצע של 17 הדגימות שהתבצעו למרחקים.
3. Telemeter\_Callback – פעולת מוד Telemeter. מציירת גרף בשידור חי, על זמן הקליטה, והמרחק שנקלט.
4. Button2D\_Callback – פעולת מוד 2D Movement. מציגה את תמונה הבקר, ביחס לשינוי המדידות שהתבצעו בחיישן התנועה.
5. Button3D\_Callback – פעולת מוד Depth Graph. מציגה הנחיות מפורטות אודות תהליך הסריקה למשתמש, ולפי בחירות המשתמש, מציגה גרף עומק בשידור חי, ולאחר סיום הסריקה, מבצעת אינטרפולציה לסריקות שנעשו ומציגה תמונת קירוב לאובייקט שנסרק.

