

אוניברסיטת בן גוריון
בית הספר להנדסת חשמל ומחשבים

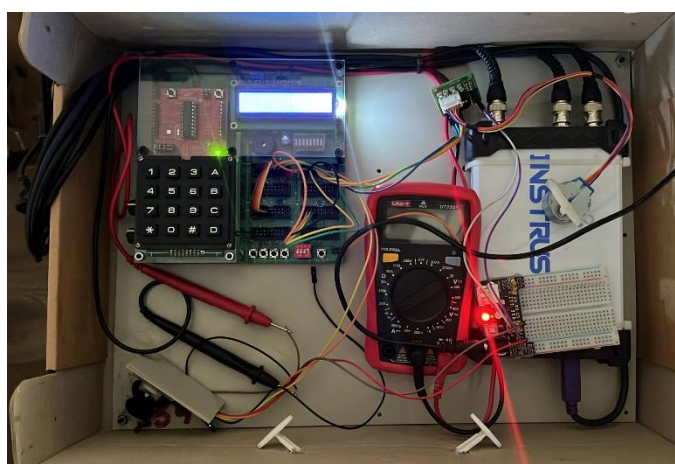
דו"ח תיעוד פרויקט גמר קורס
"מבנה מחשבים ספרתיים"
361-1-4191

שם הפרויקט: Control System of Motor-Based Machine

מגישים: מיכאל לייב, איתמר מאיר

שם המדריך: חנן ריבוא

תאריך הגשה: 01.09.2024



הגדרת ומטרת הפרויקט:

מטרת הפרויקט היא לתכנן ולממש מערכת משובצת מחשב מתקדמת המבוססת על MCU (מיקרו-בקרי) לשליטה מדויקת במכונה המופעלת על ידי מנוע צעד (Stepper Motor). המערכת נועדה לשלב שתי שיטות שליטה עיקריות:

1. שליטה ידנית מקומית באמצעות מוט היגוי אנלוגי (analog joystick).

2. שליטה מרחוק דרך מחשב אישי, תוך שימוש בתקשורת טורית.

הפרויקט שם דגש על מספר היבטים קריטיים בתכנון מערכות משובצות:

- ביצועים גבוהים תחת אילוצי זמן אמת (Real Time).

- צריכת הספק נמוכה לייעול אנרגטי.

- דיוק גבוה בביצוע פעולות ובתגובה לפקודות.

- תכנון מודולרי ושכבתי לשיפור יכולת התחזוקה והרחבה עתידית.

פרויקט הגמר משלב את הנושאים השונים שנלמדו במהלך הקורס ובמעבדות השונות.

תיאור קצר של הפרויקט:

הפרויקט מקיף מגוון רחב של היבטי פיתוח תוכנה וחומרה:

1. פיתוח תוכנה:

- כתיבת קוד ב-C/C++ הנצרב לבקרי, למימוש מערכת Embedded.

- תכנון ומימוש גרעין הפעלה מבוסס FSM (מכונת מצבים סופית) הן בצד הבקרי והן בצד המחשב.

- יישום ארכיטקטורת תוכנה שכבתית, כולל שכבות אבסטרקציה לניהול חיישנים.

- פיתוח ממשק משתמש גרפי (GUI) בצד המחשב, תוך שימוש בשפת פייתון.

2. תקשורת:

- מימוש תקשורת טורית אסינכרונית בין ה-MCU למחשב ה-PC לפי תקן RS-232.

- פיתוח פרוטוקול תקשורת לשליחה וקבלה של פקודות ונתונים.

3. בקרת חומרה:

- תכנות ה-MCU לשליטה מדויקת במנוע צעד.

- אינטגרציה של מוט היגוי אנלוגי כאמצעי קלט.

- ניהול תצוגת LCD לאינדיקציות מקומיות.

4. מודים פונקציונליים:

א. שליטה ידנית במנוע צעד:

- בקרה דינמית על זווית ה-pointer של המנוע בהיקף של 360 מעלות.

- שליטה בתדר הזזת המנוע בטווח של 50Hz – 5Hz.

ב. צייר מבוסס PC ו-Joystick:

- מימוש ממשק ציור על מסך המחשב הנשלט על ידי ה-joystick.

- תמיכה במצבי כתיבה, מחיקה, וניווט.

ג. כיול מנוע צעד:

- פיתוח אלגוריתם לכיול אוטומטי של המנוע.

- מדידה וחישוב מדויק של גודל הצעד האמיתי של המנוע.

ד. מצב Script:

- יכולת לקבל, לשמור, ולבצע קבצי script המכילים פקודות בשפה עילית.

- תמיכה בעד שלושה קבצי script שונים בזיכרון ה-FLASH של הבקר.

5. אופטימיזציה וביצועים:

- מימוש לוגיקה יעילה לעמידה בדרישות זמן אמת.

- ניהול יעיל של זיכרון מוגבל, במיוחד בצד ה-MCU.

- אופטימיזציה של זמני תגובה ודיוק בביצוע פעולות.

הפרויקט מדגיש יצירתיות בפתרון בעיות, יכולת אינטגרציה של מערכות מורכבות, והבנה מעמיקה של עקרונות תכנון מערכות משובצות. התוצר הסופי צפוי להיות מערכת רובוטית, יעילה, ומדויקת המדגימה שליטה מתקדמת במנוע צעד תוך שילוב ממשקי משתמש מתקדמים.

ביצועי החומרה והתוכנה

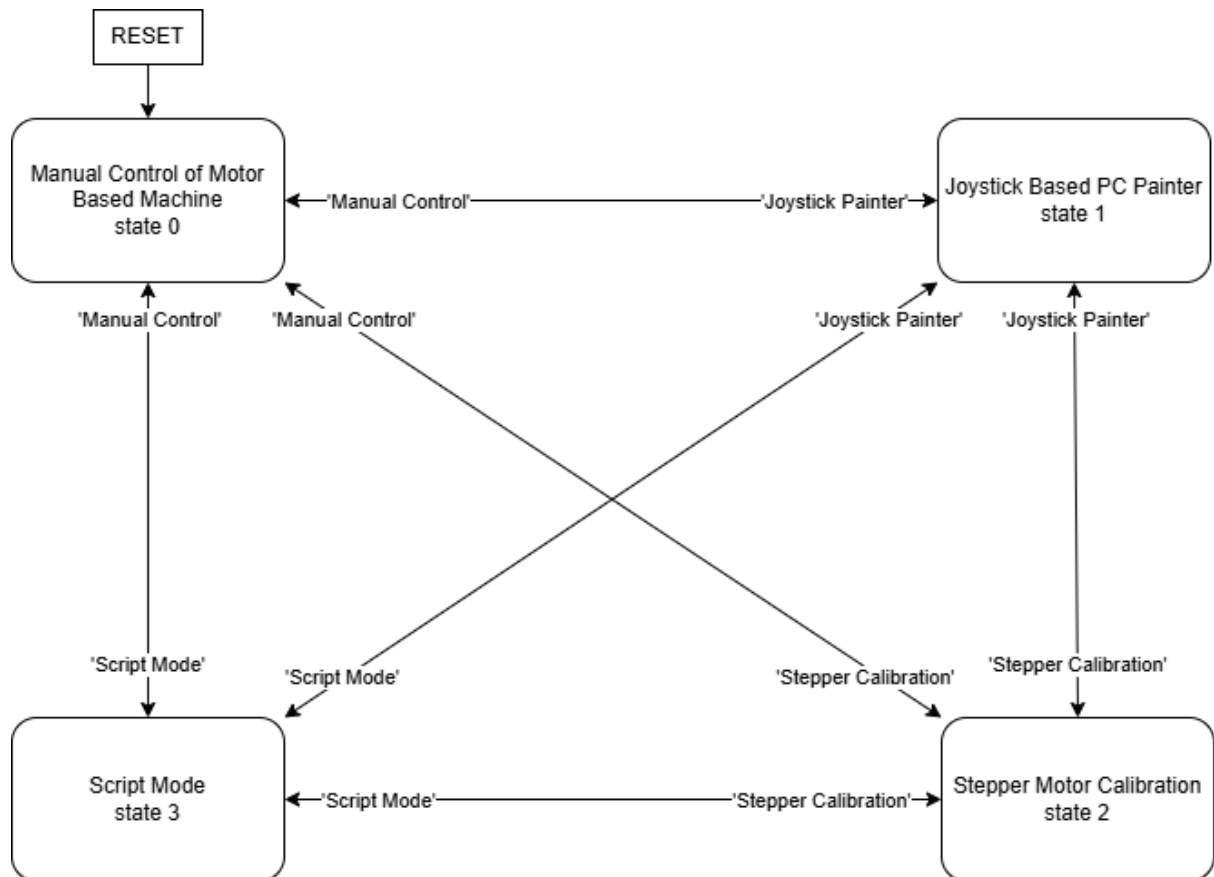
מיקרו-בקר (MCU) המבצע ייעוד מסוים עושה שימוש הן בחומרה והן בתוכנה. באופן גס ניתן לומר שהחומרה אחראית לאיסוף נתונים מסביבה חיצונית ולפעילות תקינה של הבקר, בעוד שהחלק התוכני מעבד את הנתונים ומשתמש בחומרה הנתונה לטובת המשימה.

החלק החומרתי אצלנו כולל בין היתר קלט של הג'ויסטיק באמצעות 3 קווי קלט וערכי מתח משתנים (עבור ציר X, ציר Y, לחצן על הג'ויסטיק). המערכת דוגמת את ערכי המתח של שלושת הווקטורים הנ"ל באמצעות רכיב חומרתי הנקרא ADC (ממיר מאנלוגי לדיגיטלי) ובכך ניתן לעבד את ערכי הדגימות הללו באמצעות התוכנה. בעבור הפלט, נוסף לנו מנוע צעד (Stepper motor). מתוך הבקר נכתבים מתחים ל-4 כניסות, כאשר כל כניסה מייצגת פאזה. באמצעות כתיבת המתחים ניתן להזיז את המנוע באופן הרצוי. מודול חומרתי נוסף וחשוב שנעשה שימוש באמצעותו בפרויקט זה הוא תמיכה בתקשורת UART. מודול זה מאפשר לערוך תקשורת סריאלית (טורית) בין הבקר למחשב ושליחת מידע באופן אמין. בנוסף, השתמשנו במסך LCD לטובת פלט בעבור משימות בהן נדרשנו לכך. כמובן שכמו בכל בקר יש שימוש בחומרה בסיסית (ליבה) כמו זיכרונות ומעבד.

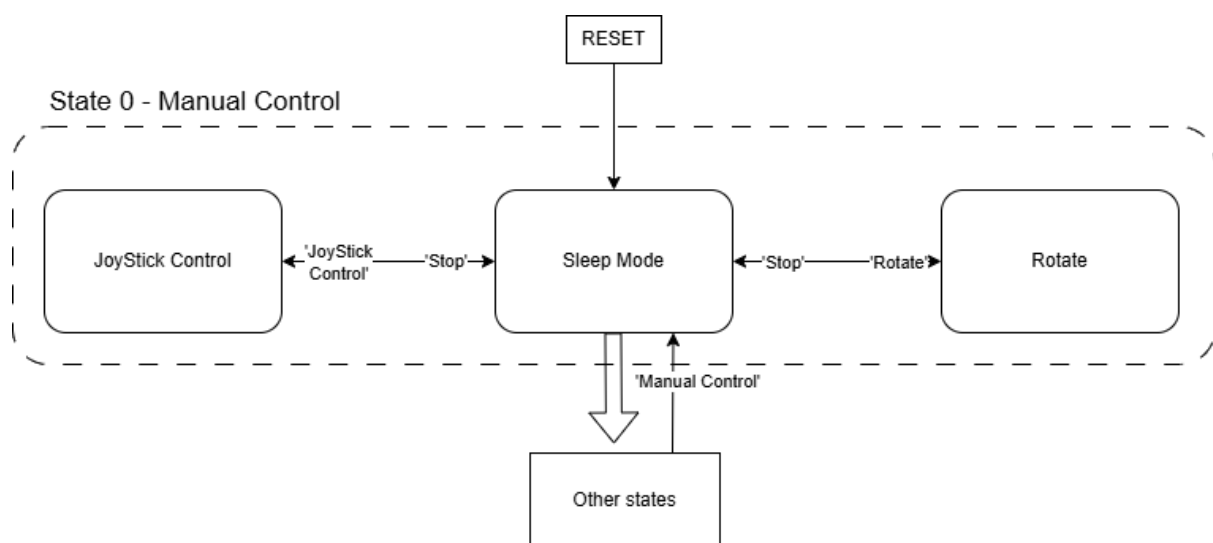
החלק התוכני עושה שימוש באופן מושכל בחומרה. בראש ובראשונה התוכנה מגדירה את ה-FSM שבאמצעותו מתנהלת כל המערכת. כמו כן, החומרה כאמור מעבדת את הנתונים המגיעים מהתוכנה לצורך ביצוע חישובים. לדוגמא, הדגימה החומרתית מהג'ויסטיק ממירה את ערכי המתח למספר בין 0 ל-1023 וכך ניתן לקבוע את הכיוון שאליו נמשך הג'ויסטיק. העיבוד התוכני מחשב את הזווית אליו מצביע הג'ויסטיק כך שניתן לגרום למנוע הצעד להצביע לאותו הכיוון. זוהי רק דוגמא קטנה הממחישה את חלוקת העבודה בין התוכנה לחומרה והאינטגרציה ביניהן.

תרשימים:

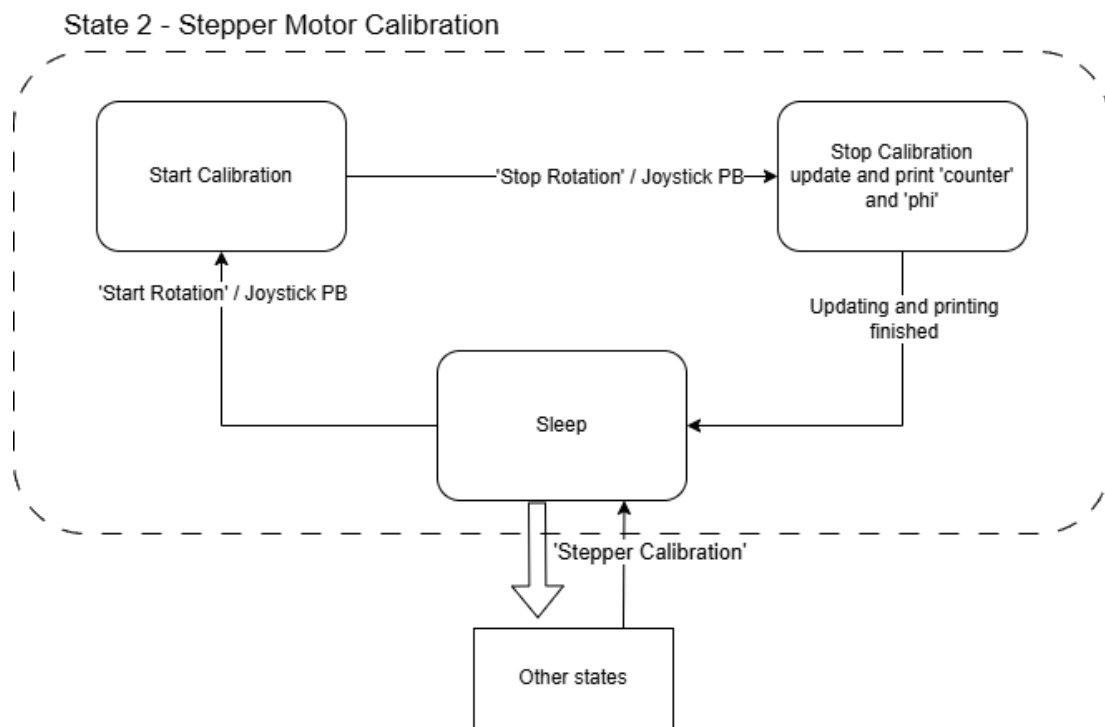
מכונת המצבים (FSM):



מבט פנימי על Manual Control:



מבט פנימי על Stepper Motor Calibration:



מעגל סכמטי:



- LCD data
- LCD control
- X,Y JoyStick
- PB0 interrupt (since JoyStick's PB doesn't work well)
- 4 phases for the stepper motor

מסקנות והצעות לשיפורים:

מסקנות:

- תכנון ראשוני לפני תחילת העבודה של המערכת הוא מאוד חשוב והכרחי.
- יעילות הקוד מאוד חשובה כאשר מדובר במשטר Hard Real Time.
- יצירת פונקציות בסיסיות וכלליות מקלה על כתיבת הקוד ומאפשרת שימוש באותה פונקציה במקרים שונים.

הצעות לייעול:

- מציאת פתרון אחר לביצוע הפונקציה GotoAngle באמצעות משתנים שהם unsigned int במקום unsigned long מאחר ו-long תופס יותר זיכרון.
- הוספת אפשרות לקביעת מהירות המנוע על ידי המשתמש.
- בפונקציה השולטת על המנוע באמצעות הג'ויסטיק: מציאת פתרון לכך שאם משנים את מיקום הג'ויסטיק לכיוון אחד תוך כדי תנועת המנוע לכיוון אחר, עצירת תנועת המנוע לכיוון הראשון ועדכון תנועתו לכיוון העדכני של הג'ויסטיק.
- בפונקציה stepper_deg להציג את הזווית באופן רציף.