

דו"ח מסכם 3 מבוא למחשבים

רועי שחמון 206564759
ניל הדר 316508332

תיאור קצר של מטלת המעבדה:

נדרשנו לצרף מצב נוסף SWstate=0x05 למטלת הדו"ח מכין בהינתן משתנה ת"ז משורשר של המשתתפים באורך 16 נדרשנו למיין את ספרות הת"ז בסדר עולה, דרך פעולה היא ע"י הגדרת מערך אינדקסים ממיון.

ראשית הגדרנו ואתחלנו:

Sortidx - מערך שבו יש 16 ערכים של אינדקסים ממיונים, שכל אחד מהם הוא מילה.
Size - מילה שמאותחלת להיות כגודל המערך הנ"ל פלוס אחד.
IDs - מחרוזת של ספרות הת"ז שכל אחד מהם הוא בייט.
P1Config# - קראנו לרוטינה שמגדירה את P1.0-P1.3 כ input
P2Config# - קראנו לרוטינה שמאתחלה את כל רגלי P2 output

: Main

התפריט הראשי בודק ע"י הפקודה bit.b באיזה מצב מתגי ה SW נמצאים בהתאם לכך קוראי לרוטינה המתאימה מ functions, שם ע"י מספר פקודות נבצע את הפעולות הבאות:
אם אנחנו במצב SWstate=0x01 נדליק את 8 הLEDים בספירה בינארית כלפי מעלה החל מערך 0 עד לערך 0xFF,
אם אנחנו במצב SWstate=0x02 נדליק את 8 הLEDים בספירה בינארית כלפי מטה החל מערך 0xFF עד לערך 0,
אם אנחנו במצב SWstate=0x04 נדליק את מערך הLEDים בצורה טורית לפי ערך המחרוזת של IDs,
אם אנחנו במצב SWstate=0x05 נדליק את מערך הLEDים בצורה טורית לפי ערך המחרוזת של IDS ממיינת.

אלגוריתם ופתרון:

מצב ראשון – אתחלנו ל0 את רגיסטר R5 וע"י mov.b שידרנו ל - P2OUT, קידמנו את R5 ע"י inc.b וקראנו לרוטינה Delay1sec כך שנוכל בעין אנושית לראות את הספירה הבינארית שמוצגת בLEDים ע"י הפקודה jmp חזרנו לשורה הראשונה בלולאה, ביצענו בדיקה האם אנחנו עדיין במצב המתאים אם כן תמשיך בלולאה, (כלומר R5 ממשיך לגדול עד לערך 0xFF וכל פעם יש השהייה של שנייה לכן נראה את הספירה בינארית המוצגת בLEDים.
מצב שני- זהה למצב המתואר הנ"ל אך הפעם טענו את R6 בערך 0xFF וביצענו dec.b לרגיסטר ושידרנו לP2OUT מצב שלישי- לפי המשתנה IDs והרגיסטר R7 שמחזיק את הכתובת של המחרוזת הצגנו את הספירה בLEDים לא לפני שנבצע sub.b #48, על מנת שנוכל לבצע המרה מספרה למספר ע"י חיסור הערך לפי טבלת ה ASCII, באופן דומה ביצענו לולאה שבודקת שאנחנו עדיין במצב הנתון מקדמת הכתובת של המחרוזת, וקוראת להשהייה.
מצב רביעי- המערך Sortidx עזר לנו באופן עקיף לדעת איזה כתובת מחזיקה את הספירה המתאימה שעלינו לשלוף מ IDs על מנת שנוכל להציג אותה על גבי הLEDים, הגדרנו ראשית רגיסטר ששומר את הכתובת של המערך של IDs שלא ילך לאיבוד כל פעם שנקדם את המצביע, כעת נכנס ללולאה print אשר תפקידה להוסיף את הערך שנמצא ב Sortidx ע"י add כדי להגיע לכתובת הדרושה אשר מחזיקה את הערך של הספירה המתאימה ב IDs באופן זהה למצב הנ"ל נחסר ב48 כדי לבצע המרה מספרה למספר במערך, לאחר מכן נשדר P2OUT ונקרא לרוטינה Delay1sec כעת נקדם ע"י incd את המצביע במערך Sortidx ונחזור ללואה של print שם קודם כל נאפס את המצביע של IDs לתחילת הרשימה. השתמשנו לשם הלולאה ברגיסטר R10 שמחזיק את אורך המארך פלוס אחד, כך שברגע שהוא יתאפס נצא מהלולאה ונבדוק שנית באיזה מצב אנו נמצאים ע"י P1IN , bit.b. באופן אלגוריתמי ע"י מערך האינדקסים ידענו לאן לפנות בזיכרון ולשלוף את הכתובת המתאימה על מנת להציג לLEDים את ספרות תעודת הזהות בסדר עולה וממיון.

גודל התוכנית: 0x00F6 בתים

זמן ריצת התוכנית: $T = 17446278 * 0.954\mu s \approx 16s$