

Preparation Report- Lab 1

Tomer Sharon 314969601

Alon Gorka 206924698

שאלות תיאורטיות:

1. PxDIR - קביעת כיוונית ערך הבקר.
PxSEL - ברירה בין מודולי החומרה המשתמשים באותה רגל בקר.
PxIN - קריא ערך המתח הלוגי ברגל הבקר.
PxOUT - קביעת ערך המתח הלוגי במוצא רגל הבקר.
2. PxDIR - נקבע ערכו ל-0. כל המבואות מוגדרים להיות כניסה.
PxSEL - נקבעים להיות כולם GPIO ולא מודולים אחרים.
PxIN - לא קשור לאתחול, קשור למה שמחובר אליהן.
PxOUT - נקבע ל-0. המוצאים נקבעים לערך מתח לוגי נמוך.

ניתן לכפות אתחול אחר לרגיסטרים האלה בקוד, אך הם נקבעים כך מתוך עקרונות העבודה הנהוגים, כדי לחסוך באנרגיה ולעבוד בצורה יעילה יותר.
3. רשום את השלבים לצורך קינפוג PORT9 למצב, O/I כאשר מבואות בעלי אנדקס זוגי במצב output ומבואות בעלי אנדקס אי-זוגי במצב input :
 - נראה שאכן ערך ברירת המחדל בPxSEL או שנכניס לו את הערך 0 ידנית.
 - כעת נבצע את הפקודה הנ"ל

```
bis.b #0xff,&P9DIR
```

```
bic.b #0xAA,&P9 DIR
```

 - נכניס את הערך '1' וכך נקבע שכיוונית הבקר תהינה לOUTPUT בכל ביטי הרגל
 - ננקה את הביטים האי-זוגיים ובכך נעביר את הביטים הללו לערך הלוגי '0' וכך יהיו במצב INPUT.
4. ניזכר שתדר מחזור השעון שלנו הוא
$$f_{MCLK} = 32 \cdot 32,768 = 2^{20} = 1,048,576 \text{ Hz}$$

c

ולכן זמן מחזור אחד הוא:

$$T_{MCLK} = \frac{1}{2^{20}} \approx 0.954 [\mu sec]$$

על מנת להגיע להשהייה של 1 [msec] נפתור את המשוואה הבאה כאשר X הוא מספר מחזורי השעון שיש להשהות:

$$T_{MCLK} * x = 1 [msec]$$

נציב נקבל

$$X = 1048.576$$
5. פסיקה היא אות חשמלי המתקבל ב-CPU מרכיב חומרה או תוכנה ומאפשר לשנות את סדר ביצוע הפקודות בתוכנית שכתבנו שלא על ידי בקרה מותנית כדוגמת הלולאה.
6. תשאול עובד עם לולאה אינסופית, אשר בודקת את מצב הלחצים לעומת פסיקה שעובדת רק אשר מופיע השינוי שנגרם ע"י לחיצת הכפתור. היתרון היחסי של התשאול על הפסיקה הוא שימוש נכון ב-CPU. (חסכון באנרגיה)

7. פסיקה חיצונית- פסיקה הנגמרת ע"י חומרה בלבד, לא תלויה בריצת התוכנית. פסיקה פנימית- פסיקה הנגרמת ע"י חומרה\תוכנה, פסיקה שניתן לחזות. פסיקת תוכנה- כאשר הדלקנו דגל בתוכנה, הפסיקה תתעורר.

8. אופני העבודה הינן פקודות LPM אשר נועדו להתאים עצמן לצרכי האפליקציה כדי להימנע מפעולות שאינן רצויות, בכך שנמנע מהבקר להגיע לאזור בזכרון שאינו מכיל פקודות רצויות, ובכך לחסוך אנרגיה. האופים השונים:

- ACTIVE: ה-CPU מופעל וכך גם כל המודולים הפריפריאליים.
- LPM0: רק ה-CPU וה-MCLK מכובים, השאר יפעלו כרגיל.
- LPM1: ה-CPU מכובה, MCLK יפעל לפי דרישה.
- LPM2: כלל השעונים וה-CPU מושבתים למעט ה-ACLK וה-DCLK.
- LPM3: כלל השעונים וה-CPU מושבתים למעט ה-ACLK.
- LPM4: כלל השעונים וה-CPU מושבתים.

9. השלבים לקנפוג הם:

```
BIC.B #0X1, &P2SEL
BIC.B #0X1, &P2DIR
BIS.B #0X01, &P2IES
BIS.B #0X01, &P2IE
```

10. Blocking- פרדיגמה זו לא מאפשרת לחלקים אחרים בקוד להתבצע עד שהן לא מסתיימות לחלוטין.

Non blocking- פרדיגמה זו מאפשרת לחלקים אחרים בקוד להפריע לה, וחוזרת לביצוע מאותה נקודה ("call back") לאחר סיום הקוד שהפריע לה.

Interrupt Driven- המעבד מקבל קריאה (פסיקה), משהה את התוכנית הנוכחית ומבצע את רוטינת הפסיקה. לאחר שסיים ישחזר את המצב וימשיך את הקוד. יש לציין שיש רוטינות שלא יפסקו מיידית (קטע קוד אטומי- לא מפריעים לו), זה נקבע לפי מצב הדגל GIE (GLOBAL INTERRUPT ENABLE).

Event Driven- התוכנית מתוכננת להתבצע בסדר מסוים שנקבע מראש ע"י היוזר או ע"י תוכנית אחרת.

מצורפת טבלת המצבים של הקוד שמימשנו:



