תוכן עניינים:

A. הקדמה: 2

B. חומר הכנה למעבדה: 2

C. שאלות הכנה: 2

D. חלק מעשי – כתיבת קוד (על גבי ערכת הפיתוח האישית מבוססת MSP430G2xx3): 3

E. צורת הגשה דוח מכין: 5

F. צורת הגשה דוח מסכם: 5

Universal Asynchronous Receiver/Transmitter (UART) Module

A. הקדמה:

בכתיבת קוד מערכת לדרישות מעבדה 4 המבוססת תקשורת טורית אסינכרונית, עליכם ליצור תקשורת בין מחשב PC ולבין שבב בקר MSP430G2553 (הנמצא בערכה פיתוח אישית). לצורך כך עליכם להגדיר בכל צד הלוקח חלק בתקשורת (צד מחשב וצד בקר) את ההגדרות המתאימות לצורך ביצוע התקשרות בניהם.
באופן פיזי, חיבור התקשורת בין המחשב לבקר נעשה דרך חיבור USB הקיים בערכה (דרכו אתם מתחברים לסביבת הפיתוח CCS לצורך פיתוח קוד המערכת – Debug mode , Active application mode). באופן כללי, ישנו צ'יפ על הכרטיס המבצע המרה מפרוטוקול תקשורת טורי USB לפרוטוקול תקשורת טורי JTAG עבור תמיכה בפעולת Debug, ובנוסף המרה לפרוטוקול טורי RS-232 עבור תמיכה בתקשורת טורית אסינכרונית עם הבקר בשימוש רכיב פריפריאלי UART. ***באופן פרטני עבור בקר מסוג MSP430G2553, דרך חיבור USB מתבצעת בפועל תקשורת מול המחשב דרך חיבור פנימי לרגלי הבקר P1.2-TX , P1.1-RX – אין לכם צורך לחבר כלום מלבד כבל USB רק זכרו שרגלי הבקר P1.2 , P1.1 תפוסים עבור התקשורת***).

1. **צד בקר** - התקשורת בצד הבקר תהיה בעזרת מודול UART, הקוד יהיה מבוסס interrupt driven בשידור וקליטה. כלומר עליכם לכתוב תוכנית שתומכת בשידור וקליטה בשימוש פסיקות בלבד (ראו באתר המודל קודים לדוגמה תחת לשונית *Personal Evaluation Kit*). **להזכירכם, עבודה בסביבת הפיתוח CCS בלבד.**

2. **צד מחשב** - בעזרת כתיבת אפליקציה על גבי המחשב המבוססת קוד בשפה עילית לבחירתכם, כגון: C#, MATLAB,Python, C++,C,JAVA, etc (***במודל נתון לכם שלושה קובצי קוד לדוגמה מבוסס Python – צד בקר + צד מחשב - כאשר התקשורת באפליקציה צד מחשב מבוססת גישת blocking בה מתבצעת האזנה ל Buffer הקלט RX עד לריקונו ושידור דרך Buffer הפלט TX עד לסיום השידור. מסיבה זו האפליקציה מבוססת מנגנון FSM***).
באופן כללי, בכל לחיצת תו במקלדת המחשב, נשלח תו מהמחשב לבקר.

B. חומר הכנה למעבדה:

1. חומר הנלמד בהרצאות + למידה עצמית – *MSP430x2xx Family User's Guide Chapter 15*.
2. הרצת הקוד לדוגמא והבנתו.
3. חידוד הנושאים בשעות הקבלה של המדריכים.

C. שאלות הכנה:

1. הסבר את אופן הפעולה של הרכיב הפריפריאלי UART ומהי מטרת שימושו.
2. הסבר את השוני בין UART ו- RS-232 וכיצד כל אחד מהם מתאים למודל שבע השכבות.
3. מהי מטרת השימוש ב- Parity Bit וכיצד מטפלת בכך המערכת.
4. הסבר את המושגים Baud Rate ו-Modulation וכיצד נקבע קצב התקשורת.
5. במצב של קליטה, כיצד קובעת המערכת את ערכו של כל ביט במידע שמתקבל.

6. הסבר ופרט את מבנה ופעולת בקר הפסיקות עבור קליטה ושידור.
7. הסבר את המושגים: Framing error, Parity error, Receive overrun error, Break condition
8. עבור 9600-8N1 ('8' – כמות סיביות מידע, 'N' - ללא סיבית זוגיות, '1' - כמות סיביות stop , קצב שידור (9600bps) ו- BRCLK=32768Hz. רשום את ערך הרגיסטרים :

UCA0CTL1, UCA0BR0, UCA0BR1, UCA0MCTL

D. חלק מעשי – כתיבת קוד (על גבי ערכת הפיתוח האישית מבוססת MSP430G2xx3):

ארכיטקטורת התוכנה בצד בקר של המערכת נדרשת להיות מבוססת ***Simple FSM*** המבצעת קטע קוד השייך לאחד ממצבי המערכת בהינתן בקשת פסיקה המגיעה ממחשב PC לבקר דרך ערוץ התקשורת למודל UART.

קוד המערכת נדרש להיות מחולק לשכבות כך שהוא יהיה נייד (portable) בקלות בין משפחות MSP430x2xx , MSP430x4xx ע"י החלפת שכבת ה- BSP בלבד.

1. טרם שלב כתיבת הקוד נדרש לתכנן ולשרטט גרף לכל אחת משתי דיאגרמות FSM הבאות (חלק מדו"ח מכין):
   - גרף של ארכיטקטורת המערכת בצד בקר (כמתואר בסעיף D5)
   - גרף של אפליקציית התקשורת בצד מחשב (כמתואר בסעיף A2)
2. אסור לבצע השהייה ע"י שימוש ב poling למעט עבור debounce ברוטינת שירות של בקשות פסיקה בגין לחצנים.
3. **בתחילת התוכנית, הבקר נמצא במצב שינה.**
4. חיבורי חומרה נדרשים:
   - חיבור לד RGB: **R** – P2.2, **G** – P2.1, **B** – P2.0
   - חיבור Buzzer לרגל P2.4
   - חיבור פוטנציומטר (נגד משתנה) לרגל P1.3
   - מסך LCD נדרש לחבר את D7-D4 לרגליים P1.7-P1.4 בהתאמה (אופן עבודה של ה- LCD בארבע סיביות של מידע) + שלושת קווי הבקרה של ה- LCD לרגליים P2.7, P2.6, P2.5
5. עליכם לבצע תקשורת בין הבקר ולבין ה-PC, תחת הדרישות הבאות:
   - בלחיצת מספר במקלדת המחשב של השורה המתאימה בתפריט, תתבצע הפעולה המתאימה בבקר
   - להלן הגדרת התפריט להדפסה על מסך המחשב
   - כל שורה בתפריט מהווה מצב במערכת כך שהמערכת מכילה שמונה מצבים שונים כאשר ערך כל מצב הוא מספר שורה בתפריט. **הקשתות בגרף המערכת בגינן מתקבלת החלטה למעבר ממצב אחד למצב אחר נעשית בשכבת ה- HAL ב ISR של RX עקב קלט מידע (מידע נקלט מכונה Command לצורך ברירת מצב המערכת). בכניסה למצב, המידע הנקלט משמש כ Data למצב הנמצא בביצוע. בהתאם לאמור לעיל, יש צורך לסווג את המידע הנקלט למידע מסוג Command ולמידע מסוג Data (כתחליף לשימוש בלחצנים כפי שבצעתם בניסויים LAB1-LAB3).**

Menu

1. Blink RGB LED, color by color with delay of X[ms]
2. Counting up onto LCD screen with delay of X[ms]
3. Circular tone series via Buzzer with delay of X[ms]
4. Get delay time X[ms]:
5. Potentiometer 3-digit value [v] onto LCD
6. Clear LCD screen
7. Show menu
8. Sleep

הסבר סעיפי התפריט:

נתון שערך ברירת המחדל של X הוא 500ms (כלומר X=500).

1. נדרש להבהב את לד RGB צבע אחר צבע עם השהיה של X בין כל החלפת צבע.
2. מנייה מעלה בעזרת משתנה מטיפוס *int* תוך שמירת ערך המנייה במעבר בין מצבים (ראו הערה בסעיף 6)
3. השמעת צלילים דרך ה Buzzer בצורה מעגלית מתוך הסדרה הבאה עם השהיה של X בין כל החלפת tone . tone series = {1kHz, 1.25kHz, 1.5kHz, 1.75kHz, 2kHz, 2.25kHz, 2.5kHz}
4. קליטת ערך X מהמשתמש (היחידות של המספר X הן ms). בלחיצת 4 בתפריט נדרש לקלוט מהמשתמש מחרוזת ספרות (המייצגת את X) כאשר תו הסיום ENTER .
5. מדידת ערך מתח מרגל Potentiometer **בצורה דינאמית**, בדיוק שלוש ספרות (ביחידות של volt) והדפסת הערך הנמדד למסך ה LCD (עדכון ערך במסך LCD יעודכן רק במיקום הרלוונטי).
6. ניקוי מסך LCD + אתחול משתנה ערך המנייה של סעיף 2 לערך אפס
7. הצגת התפריט על גבי מסך צד מחשב
8. כניסה למצב שינה (המוגדר ב idle state)

E. **צורת הגשה דוח מכין:**

- הגשת מטלת דוח מכין תיעשה ע"י העלאה למודל של תיקיית zip מהצורה **id1_id2.zip** (**כאשר id1 < id2**), **רק הסטודנט עם הת"ז id1 מעלה את הקבצים למודל.**
- התיקייה תכיל את **שלושת** הפריטים הבאים בלבד:
  - ✓ **קובץ pre_lab$_x$.pdf** – מכיל תשובות לחלק תיאורטי דו"ח מכין
  - ✓ תיקייה בשם **CCS** - מכילה שתי תיקיות, אחת של קובצי source (קבצים עם סיומת ***.c**) והשנייה של קובצי header (קבצים עם סיומת ***.h**).
  - ✓ תיקייה בשם **PC_side** - המכילה קובצי מקור של אפליקציית צד מחשב + קובץ ReadMe המתאר בקצרה מה תפקיד כל קובץ מקור במימוש האפליקציה.

F. **צורת הגשה דוח מסכם:**

- הגשת מטלת דוח מכין תיעשה ע"י העלאה למודל של תיקיית zip מהצורה **id1_id2.zip** (**כאשר id1 < id2**), **רק הסטודנט עם הת"ז id1 מעלה את הקבצים למודל.**
- התיקייה תכיל את **שלושת** הפרטים הבאים בלבד:
  - ✓ **קובץ final_lab$_x$.pdf** – מכיל תיאור והסבר לדרך הפתרון של מטלת זמן אמת.
  - ✓ תיקייה בשם **CCS** - מכילה שתי תיקיות, אחת של קובצי source (קבצים עם סיומת ***.c**) והשנייה של קובצי header (קבצים עם סיומת ***.h**).
  - ✓ תיקייה בשם **PC_side** - המכילה קובצי מקור של אפליקציית צד מחשב + קובץ ReadMe המתאר בקצרה מה תפקיד כל קובץ מקור במימוש האפליקציה.

בהצלחה