

Table of Contents

2	A <u>חומר עזר:</u>
2	B <u>שאלות הכנה תיאורטיות:</u>
3	C <u>תיאור קוד לדוגמא:</u>
4	D <u>חלק מעשי – כתיבת קוד מערכת פורטאבילי:</u>
5	E <u>צורת הגשה דוח מכין:</u>
5	F <u>צורת הגשה דוח מסכם:</u>

דו"ח מכין מס' 5 – ניסוי Basic and Advanced Timers

A. חומר עזר:

❖ נושאי המעבדה:

בניסוי מעבדה זה נעסוק בנושא Timer. בבקר אתו אנו עובדים KL25Z128VLK4 ישנם 4 מודולים שונים של Timer המיועדים לצרכים שונים. אנו נתמקד בניסוי המעבדה במודולים TPM ו-PIT.

1. Chapter 31 - Timer/PWM Module (TPM)

משמש ליצירת time interval interrupts, PWM, output compare, input capture.

2. Chapter 32 - Periodic Interrupt Timer (PIT)

משמש ליצירת time interval interrupts

3. Chapter 33 - Low-Power Timer (LPTMR) (בגדר העשרה – לא כלול בניסוי)

4. Chapter 34 - Real Time Clock (RTC) (בגדר העשרה – לא כלול בניסוי)

❖ חומר הכנה למעבדה:

1. בקובץ כרטיס הבקר "FRDM-KL25Z User's Manual" לקרוא את

הפרקים 31,32 הרלוונטיים בספר

2. להפעיל את הקוד לדוגמא ולהבין אותו (הקוד הנתון אינו מחולק לשכבות אלא דוגמה פונקציונאלית).

3. על בסיס הנ"ל לבצע את מטלת המעבדה.

B. שאלות הכנה תיאורטיות:

מודול TPMx :

- (1) הסבר את ייעודו של טיימר זה ולאיזה אפליקציות הוא מיועד.
- (2) פרט והסבר את אופני הפעולה של הטיימר.
- (3) הסבר את אופני הפעולה עבור input capture ומה הצורך בתכונה זו.
- (4) הסבר את אופני הפעולה עבור output compare ומה הצורך בתכונה זו.
- (5) הסבר את סוגי הפסיקות במודול זה ומה מטרת כל סוג פסיקה
- (6) הסבר את סוגי אותות PWM אותם ניתן להפיק במוצא הבקר ומה ההבדל ביניהם.

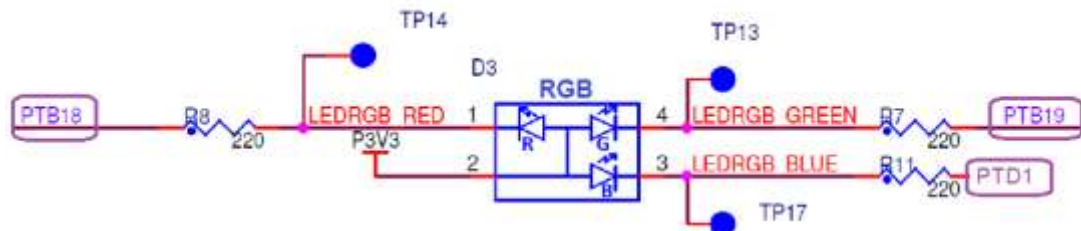
מודול PIT :

- (1) הסבר את ייעודו של טיימר זה ולאיזה אפליקציות הוא מיועד.
- (2) פרט והסבר את אופני הפעולה של הטיימר.
- (3) לצורך יצירת event כל 1msec מהם שלבי הקינפוג.

C. תיאור קוד לדוגמא:

הערה: קוד לדוגמה מכיל עבודה מול הרכיב הפריפריאלי בלבד (אינו מחולק לשכבות ואינו מבוסס פרדיגמת תכנון simple FSM כפי שנדרש מכם בביצוע המשימה).

נחבר לחצן לרגל הבקר PTD7 בצורת pull-up. בהתחלה הבקר נמצא במצב שינה והלד RGB כבוי. בלחיצה ראשונה לד RGB ידלוק ב-7 צבעים (לפי הסדר: כחול, אדום, ורוד, ירוק, תכלת, כתום, לבן) בהירות מקסימאלית **צבע אחר צבע** בהפרש של 1sec. בכל לחיצה נוספת על הלחצן רמת בהירות הצבעים תעלה עד למקסימום ולאחר מכן תתחיל מההתחלה בצורה מחזורית, נחלק את בהירות ההארה ל-5 רמות, כך נוכל לשלוט על בהירות הלדים. מעבר לתחילת התוכנית ניתן ללחוץ על לחצן RESET. עוצמת הבהירות נשלטת ע"י אות מתח PWM הנופל על הלד והנגד שבטור (נקנפג את כניסות RGB ל-PWM). הלד והנגד מרגישים מתח DC ממוצע (ולא את המתח הרגעי) לפי הנוסחה $V_{avg} = 3.3v \cdot DutyCycle$. כאשר $DutyCycle \in [0, 1]$. האיור הבא מתאר את חיבור RGB led בכרטיס הבקר.



RGB LED	KL25Z128	Pin Alternation
Red Cathode	PTB18	ALT3 = TPM2_CH0
Green Cathode	PTB19	ALT3 = TPM2_CH1
Blue Cathode	PTD1	ALT4 = TPM0_CH1

הסבר למימוש התוכנית לדוגמא:

בעזרת מודול TPM, ניצור בכל אחד מרגלי הלדים PTB18, PTB19, PTD1 את PWM עם DutyCycle משתנה לצורך הדלקת לד RGB צבע אחר צבע. החלפת הצבעים תיעשה אחת לשנייה בשימוש מודול PIT. אחת לשנייה ניכנס לרוטינת PIT ונשנה את צבע לד RGB. בכל לחיצה על לחצן PTD7 ניכנס לרוטינת פסיקת ה-I/O כדי לשנות את עצמת ההארה.

קובצי source ו-header של הקוד לדוגמא נמצאים במודל בתיקיית LAB3 - Code Example

D. חלק מעשי – כתיבת קוד מערכת פורטאבילי:**להלן תיאור צורת החיבורים הנדרשת בכרטיס ההרחבה לבקר KL25Z:**

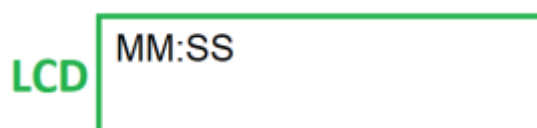
- הלחצנים PB2 - PB0 מחוברים לרגלי הבקר P1.0 – P1.2 בהתאמה
- כניסה אנלוגית נדרש לחבר לרגל הבקר P1.3 (כניסה A3)
- מסך LCD נדרש לחבר את D7-D4 לרגליים P1.7-P1.4 בהתאמה (אופן עבודה של ה- LCD בארבע סיביות של מידע) + שלושת קווי הבקרה של ה- LCD לרגליים P2.7, P2.6, P2.5 (קוד עבור LCD נתון במודל, עליכם לעדכן לצרכיכם).
- מוצא Generator מחובר לרגל P2.4 במצב של (Input capture)
- רגל P2.2 במצב של Output compare (PWM) מחוברת לערוץ הסקופ CH2 ול- Buzzer (במקביל)
- **הבהרה:** מתח Vcc בערכת הפיתוח האישית נע בין 3.5v-3.65v (בשונה מערכת הפיתוח במעבדה Vcc=3.3v)
- **הוא תלוי לאיזה מחשב מחוברת ערכת הפיתוח (קיימת שונות של מתח ה- Vcc בחיבור בין מחשבים שונים).** כדי לבדוק את ערך Vcc במדויק, זאת ניתן למדוד מהי רמת מתח המוצא של ה- '1' לוגי מאחד הפורטים.
- ארכיטקטורת התוכנה של המערכת נדרשת להיות מבוססת *Simple FSM* (כמתואר בדו"ח מכין 1) המבצעת אחת מתוך ארבע פעולות בהינתן בקשת פסיקה חיצונית של לחיצת לחצן מתוך שלושת הלחצנים.
- קוד המערכת נדרש להיות מחולק לשכבות כך שהוא יהיה נייד (portable) בקלות בין משפחות שונות של הבקר ע"י החלפת שכבת ה- BSP בלבד.
- כתיבת פונקציות ה- driver של ה- LCD צריכות להיות ממוקמות ב- HAL בעוד שפונקציה לכתיבת מחרוזת המבוססת עליהן צריכה להיות ממוקמת בשכבת ה- API.
- טרם שלב כתיבת הקוד נדרש לשרטט גרף דיאגרמת FSM **מפורטת** של ארכיטקטורת התוכנה של המערכת ולצרפה לדו"ח מכין. המצבים אלו הצמתים והקשתות אלו המעברים ממצב למצב בגין בקשות פסיקה.
- **משלב זה ואילך, אסור לבצע השהייה ע"י שימוש ב- polling למעט עבור debounce ברוטינת שירות של בקשות פסיקה בגין לחצנים.**

להלן דרישת מצבי המערכת:(state=idle=0):

בלחיצת RESET הבקר נמצא/חוזר למצב שינה (Sleep Mode).

(state=1) PB0 על לחצן:

נדרש לבצע על גבי מסך LCD שעון עצר לאחר של דקה במניית שניות (המצב ההתחלתי בכניסה למצב הוא 01:00), הקפידו על רמת הדיוק המנייה ועל מיקום המנייה על מסך ה- LCD כמופיע באיור הבא:

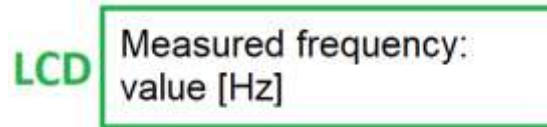


המצב מוגדר להסתיים בהגעה לערך מנייה 00:00, מצב זה נדרש לאפשר לחתוך ע"י שאר הלחצנים, המנייה חייבת להתבצע בשימוש פסיקות PITx לבחירתכם מתוך שיקולי תכנון

בלחיצה על לחצן PB1 (state=2):

נדרש לממש counter (מונה תדר) למדידת תדר אות שעון חיצוני המוזן ממחולל האותות לרגל הבקר P2.4 (ובערכת הפיתוח במעבדה P2.3) את ערך התדר הנמדד יש להציג על גבי מסך LCD, לפי הפירוט הבא:

✓ הצגת התדר ביחידות של Hz בצורה **דינאמית** (ללא הצגת היסטוריית המדידות), כאשר ערך המדידה מתעדכן נדרש לכתוב רק לשדה value ולא לרענן את כל המסך.



- ✓ ערך התדר של האות הנמדד יהיה בתחום של $f \in [20\text{Hz}, 20\text{kHz}]$
 - ✓ מרחק ברמת דיוק של תדר שלם.
 - ✓ סינון רעשים (כלומר, במידה והתדר לא משתנה המרחק הנמדד לא ישתנה).
- המצב מוגדר להסתיים בלחיצה על לחצן המשויך למצב אחר. המימוש נדרש להיות מבוסס אופן עבודה Input Capture מבוסס רכיב פריפריאלי TPMx.
- לפני חיבור המחולל לערכה, וודא שאות השעון במוצא המחולל הוא גל ריבועי עם ערכי קצוות 0v-3v.
- הקפד לחבר את המחולל בקוטביות נכונה, בננה שחורה ל- GND ובננה אדומה לפין הרצוי.

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 שני הפרטים הבאים בלבד:
 - ✓ קובץ pre_labx.pdf – מכיל תשובות לחלק תיאורטי דו"ח מכין
 - ✓ תיקייה בשם CW - מכילה שתי תיקיות, אחת בשם Sources של קובצי source (קבצים עם סיומת *.c), והשנייה בשם Project_Headers של קובצי header (קבצים עם סיומת *.h).

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 שני הפרטים הבאים בלבד:
 - ✓ קובץ final_labx.pdf – מכיל תיאור והסבר לדרך הפתרון של מטלת זמן אמת.
 - ✓ תיקייה בשם CW - מכילה שתי תיקיות, אחת בשם Sources של קובצי source (קבצים עם סיומת *.c), והשנייה בשם Project_Headers של קובצי header (קבצים עם סיומת *.h).

בהצלחה