

תוכן עניינים:

פרק 7 : ADC – Analog to Digital Converter - ממיר מאנלוגי לספרתי

2	7.1 ממיר ADC בשיטת הקירוב הרציף
7	7.2 מתח ייחוס
7	7.3 SAR CLOCK - פולסי השעון ל SAR
7	7.4 איך להתחיל המרה ?
8	7.5 – מדידת מתח הפרשית ומדידה יחסית לאדמה - Single Ended ...
10	7.6 חלון השוואה – חלון משווה - Window Comparator
11	7.7 - Right and Left Justification - יישור לימין ולשמאל
12	7.8 מגבר בר תכנות (לא קיים ב C8051F380 !!)
12	7.9 מגבר הפרש מתח גבוה HVDA (לא קיים ב C8051F380 !!)
13	7.10 מאפייני ADC
16	7.11 ממיר ה ADC ברכיב C8051F380 - ADC0
18	7.12 הרגיסטרים של ADC0
24	7.13 מתח הייחוס VREF - ל ADC
26	7.13.1 דוגמאות עבור רגיסטר REF0CN
28	7.14 תוכנה להפעלת הממיר
30	7.15 דוגמאות להפעלת הממיר
34	7.16 ביבליוגרפיה

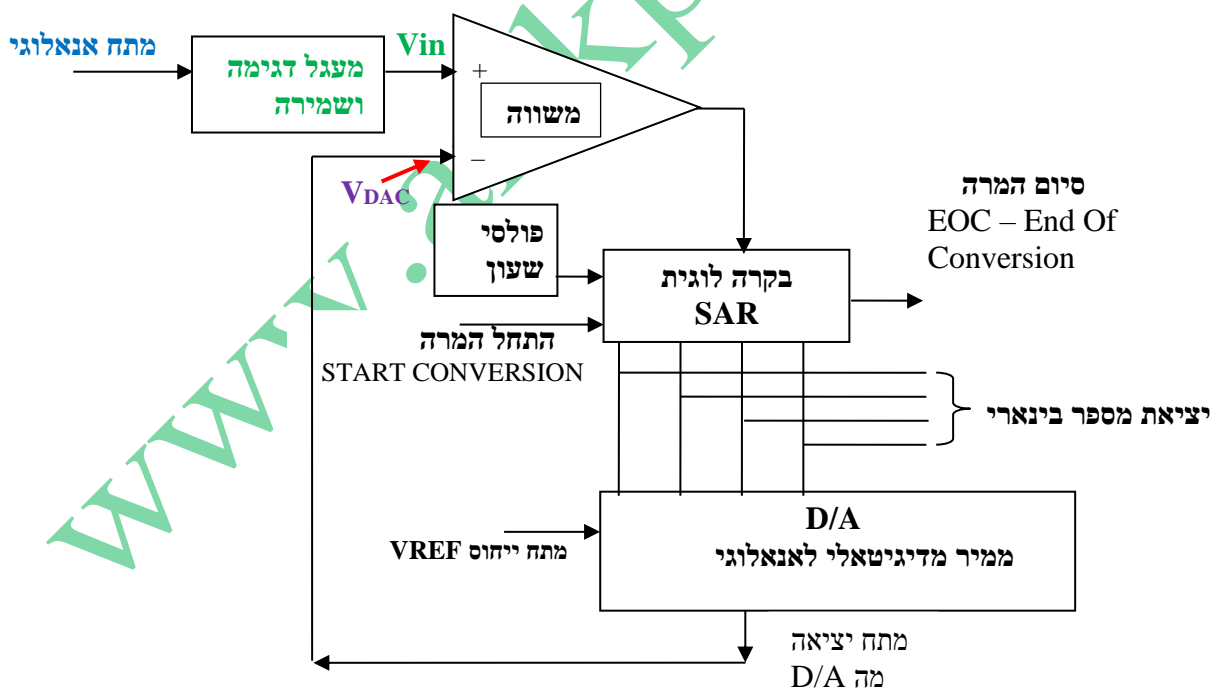
פרק 7 : ADC – Analog to Digital Converter - ממיר מאנלוגי לספרתי

חיישנים רבים העוסקים בגדלים פיזיקאליים כמו טמפרטורה, לחות, לחץ, מהירות וצמיגות נותנים ביציאה שלהם אותות אנלוגיים שיש להפוך אותם לאותות דיגיטאליים שיתאימו לעבודה עם מחשב. דוגמא פשוטה לצורך להפוך אות אנלוגי לדיגיטאלי היא תוכנית ההקלטה במחשב (נגן המדיה) ההופכת את אות השמע האנלוגי לדגימות בדידות דיגיטאליות. תפקידו של מעגל ADC - Analog to Digital Converter - ממיר מאנלוגי לדיגיטאלי – להפוך (להמיר) אות אנלוגי למספר בינארי (דיגיטאלי).

קיימים מספר סוגים של ממירי ADC. נסביר את הממיר העובד בשיטת הקירוב הרציף – SAR - Successive Approximation Register. הממיר ברכיב C8051F380. הממיר ברכיב הוא של 10 ביטים.

7.1 ממיר ADC בשיטת הקירוב הרציף - SAR - Successive Approximation Register

ב SAR ADC מתבצעת סדרה של השוואות כדי לקבוע את התוצאה עבור כל ביט של ההמרה. מכאן ש SAR ADC צריך לפחות $N+1$ פולסי שעון להמיר את מתח הכניסה האנלוגי למספר דיגיטאלי כאשר N הוא מספר הביטים של הממיר. האיור הבא מתאר את עיקרון ההמרה בממיר SAR. נדגים את עיקרון פעולתו של ממיר בן 4 ביטים (ומההסבר יהיה קל להבין גם את פעולתו של ממיר בעל 10 ביטים). הממיר מתבסס על פעולה של Digital to Analog Converter – D/A – ממיר מדיגיטאלי לאנלוגי.



איור 1 – עקרון ממיר העובד בשיטת קירוב מוצלח – Successive Approximation עם 4 ביטים.

המתח אנלוגי שרוצים להמיר נכנס בצד שמאל. מתח זה טוען קבל במעגל הדגימה והשמירה. מצב זה שבו הקבל נטען למתח הכניסה נקרא מצב **עקיבה – TRACK**. כאשר מבצעים דגימה ושמירה מעבירים את המתח שעל הקבל לכניסה העוקבת של המשווה ומתח זה לא ישתנה במהלך ההמרה. מתח זה נקרא V_{in} באיור.

המשווה מקבל שני אותות. האחד הוא V_{in} והשני הוא מתח ממיר דיגיטאלי לאנלוגי V_{DAC} . אם מתח הכניסה V_{in} גבוה ממתח הממיר - V_{DAC} - אז ביציאת המשווה יש '1' ולהפך.

הממיר דיגיטאלי לאנלוגי – D/A - מקבל מספר דיגיטאלי בכניסתו מהמערכת. הלוגית לקירוב רציף והוא מוציא מתח אנלוגי למשווה.

בקרה לוגית SAR – מערכת לוגית המנהלת את תהליך ההמרה ושומרת את התוצאה ברגיסטר ה- **Successive – SAR Approximation Register**.

Vref - מתח הייחוס הקובע את המתח המקסימאלי שיכול להוציא ה D/A.

כאשר הממיר מקבל פקודת "התחל המרה" - **START CONVERSION** - דוגם מעגל הדגימה את מתח הכניסה ומעביר מתח זה לכניסה העוקבת של המשווה.

מעגל הבקרה הלוגי נותן מספר ל D/A להוציא חצי מהמתח המקסימאלי V_{ref} במקרה שלנו 1000.

מעגל הבקרה הלוגי שומר '1' ב SAR אם מתח הכניסה גבוה ממתח הממיר D/A או '0' אם מתח הכניסה נמוך מהממיר. הביט שנשמר הוא ביט ה MSB של המספר הבינארי שיצא מהממיר בסוף תהליך ההמרה.

בשלב השני הבקרה הלוגית מעבירה מספר ל D/A לפי הביט שב MSB. הוא יוציא $\frac{3}{4}$ מהמתח המקסימאלי שלו (אם בביט ה MSB היה '1' כלומר 1100) או $\frac{1}{4}$ מהמתח המקסימאלי שלו (אם בביט היה '0' כלומר 0100).

לפי יציאת המשווה (1 או 0) שוב נשמר (בהתאמה) 1 או 0 בביט השני שב SAR.

התהליך חוזר על עצמו על עוד 2 הביטים הבאים של המספר. בסיום ההמרה מוציאה הבקרה הלוגית הודעה על סיום המרה - **EOC** - **End Of Conversion**.

קיימים ממירים שהודעת סיום ההמרה נקראת **DRDY** - **Data Ready** האומרת שיש נתון מספרי המוכן לקריאה מהממיר. בממירים אחרים זה נקרא **BUSY** וההדק אומר שהממיר עסוק בהמרה ובסיום ההמרה המצב בהדק זה מתהפך.

השם קירוב רציף או הערכה מוצלחת נובע מכך שמעגל הבקרה הלוגית שולח מספר בינארי אל ה DAC (המוציא מתח אל המשווה) ובודק האם מתח הכניסה גבוה או נמוך ממנו ועושה את ההתאמה המתבקשת.

תהליך בדיקת כל הביטים ושמירתם ברגיסטר ה SAR קיבלו את השם "ארכיטקטורת SAR".

מערכת הבקרה הלוגית פועלת בעזרת פולסי שעון. בכל פולס שעון מתבצע התהליך הבא : **א.** הוצאת מספר אל ה D/A. **ב.** בדיקת מצב המשווה **ג.** שמירת הערך לביט המתאים ב SAR.

$$T_{conv} = (N+1) \cdot T_{clk}$$

בממיר בן N ביט נדרשים N+1 פולסי שעון לבצע את ההמרה. זמן ההמרה הוא :

כאשר T_{conv} הוא זמן המרה ו T_{clk} הוא המחזור של פולסי השעון.

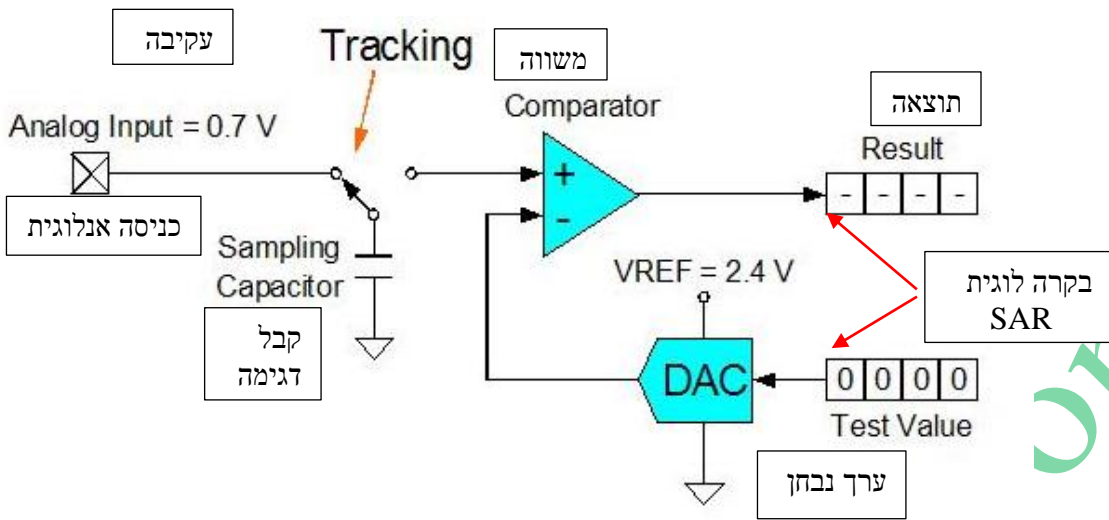
מכאן : זמן ההמרה תלוי ב (כמות הביטים של הממיר + 1) ובזמן המחזור של תדר השעון.

בסעיף הבא נתאר את פעולת הממיר בן 4 הביטים בהרחבה בעזרת דוגמאות מספריות בעזרת האיורים הבאים:

לשם כך נניח שמתח הייחוס הוא $V_{REF} = 2.4V$ ומתח הכניסה האנלוגי שנכנס להמרה הוא $0.7V$.

7.1.1 עקיבה - TRACKING

בתחילת התהליך מתבצעת עקיבה אחרי המתח האנלוגי בכניסת . מתח זה טוען את הקבל הנקרא Sampling Capacitor – קבל דגימה. בשלב זה הבקרה הלוגית SAR איננה מפעילה המרה. (בהמשך האיורים נוותר על הצגת מלבן הבקרה הלוגית SAR) .



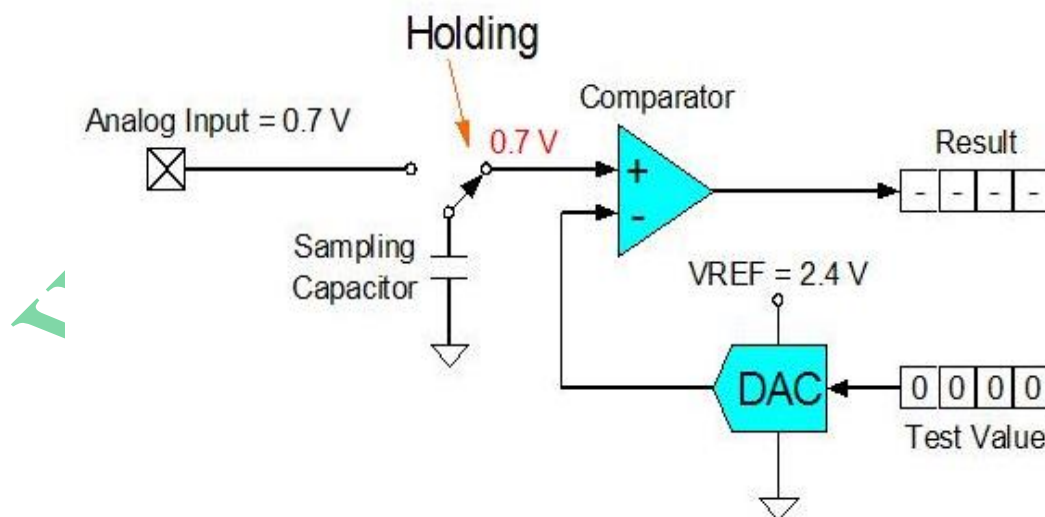
איור 2 : מצב עקיבה TRACKING

לכל SAR יש זמן עקיבה משלו, שתלוי במהירות זמן הטעינה ונקבע על ידי קיבול קבל הדגימה, והתנגדות הכניסה של ההדק הכניסה האנלוגית.

כאשר תגיע פקודת התחל המרה – Start Conversion המתג האלקטרוני יעביר את מתח הקבל לכניסה העוקבת של המשווה – Comparator .

7.1.2 - Sample and Hold – דגימה ואחזקה.

באיור הבא מתואר המעבר של המפסק לכניסת המשווה. זהו מצב החזקה שבו מתח הקבל נשמר ואיננו משתנה. הדגימה מתבצעת על מתח זה. מכאן מתחיל תהליך ההמרה.



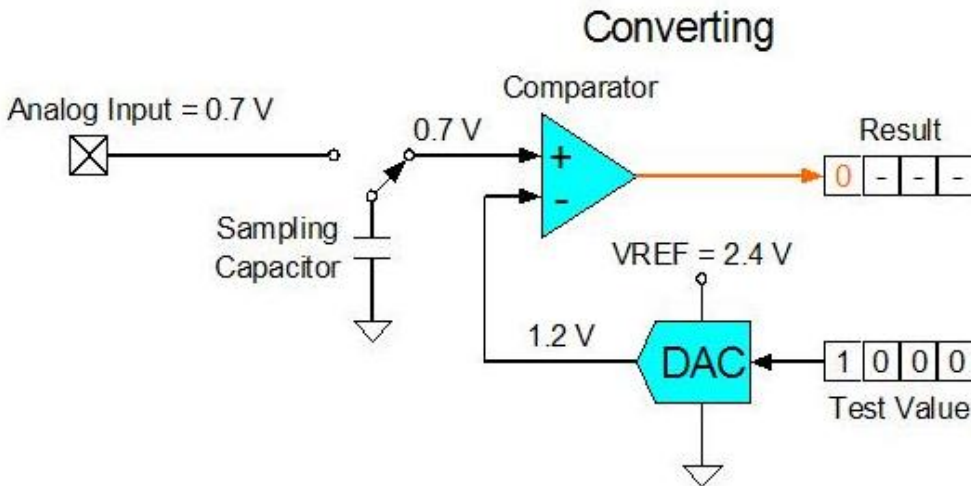
איור 3 : מצב החזקה

7.1.3 השוואה והמרה ראשונה

בתהליך ההשוואה וההמרה מתבצעת סדרה של השוואות בין מתח הקבל בכניסה האנלוגית העוקבת (+) של המשווה ובין ערך המתח בכניסה ההופכת (-) שמגיע מממיר ה DAC. אם מתח הכניסה ב + של המשווה גבוה מהמתח ב - המשווה ברוויה החיובית שלו ויוצא '1'. אם מתח ה - גדול ממתח ה + המשווה ברוויה השלילית ויוצא 0.

יציאת המשווה נרשמת בביט המתאים למספר ההמרה ברגיסטר ה Result (מביט 1 של ה MSB ועד ביט 4 של ה LSB).

האיור הבא מתאר את ההמרה הראשונה וכתובת תוצאת ההשוואה ב Result:



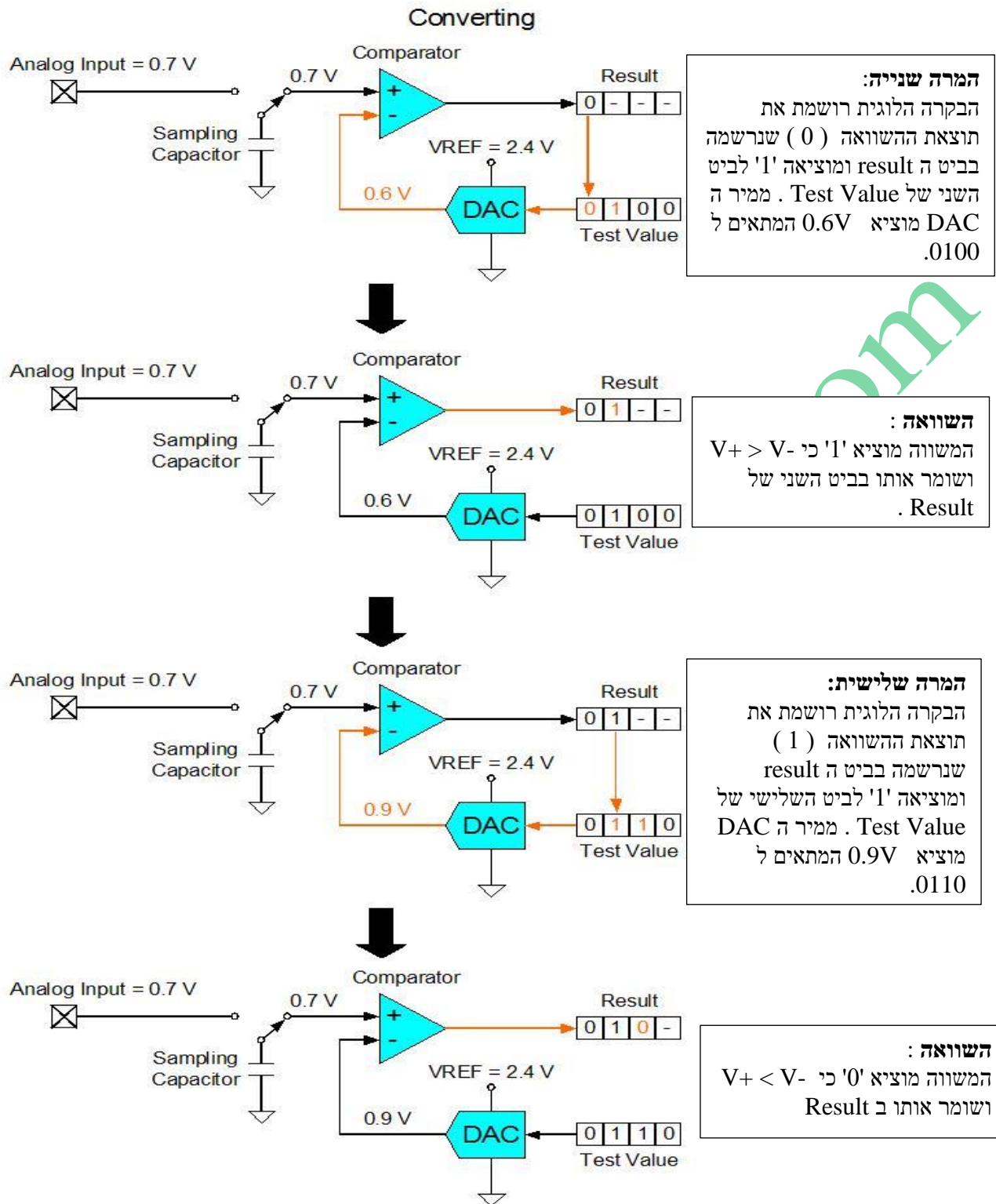
איור 4 : המרה ראשונה.

הבקרה הלוגית SAR שמה ב Test Value את הערך 1 בביט ה MSB של . לממיר ה DAC מגיע הערך 1000. ממיר ה DAC מוציא אל המשווה 1.2V (חצי מ VREF). היות ו $V_+ < V_-$ המשווה מוציא '0' שנרשם בביט הגבוה של התוצאה - RESULT.

7.1.4 המרות על הביטים הבאים

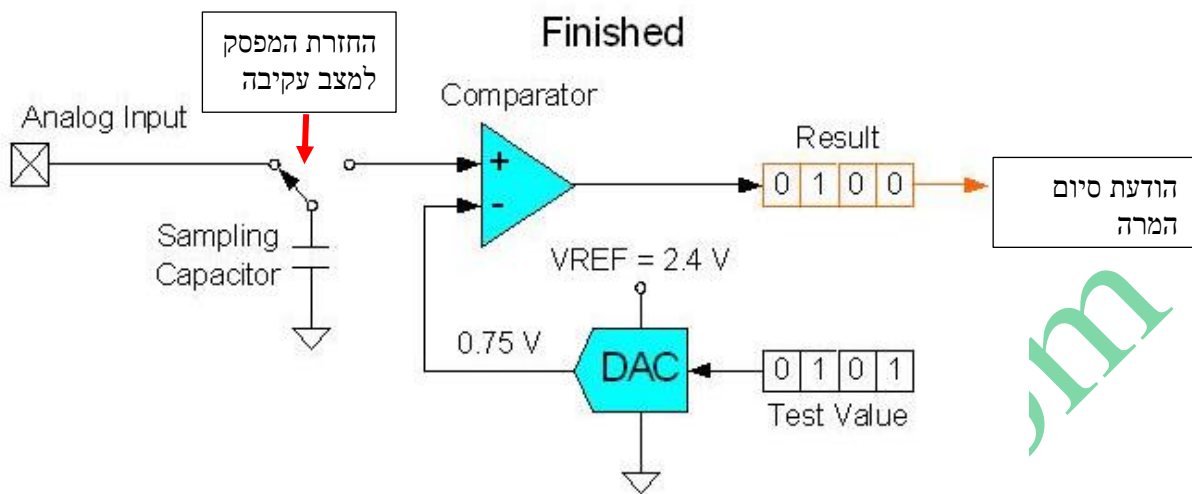
האיורים הבאים מתארים את המשך תהליכי ההמרה וההשוואה עבור הביטים 2, 3 ו 4.

באיור רואים שמערכת הבקרה הלוגית SAR מעבירה את הביט שב Result אל הביט התואם ב Test Value ואז מעבירה לביט השני ב Test Value את הערך 1. ה DAC מוציא מתח היחסי למספר 0100 ושוב מתבצעת השוואה והתוצאה של ההשוואה נרשמת ב Result. התהליך חוזר על עצמו עבור כל הביטים של הממיר (בדוגמה כאן נותרים עוד ביטים 3 ו 4).



איור 5 : המרות לביטים השני והשלישי

בהמרה האחרונה הבקרה הלוגית SAR רושמת את ה 0 שבביט השלישי ב Result בביט המתאים ב Test Value ושמה 1 בביט הרביעי של ה Test Value. האיור הבא מתאר את ההמרה וההשוואה האחרונה והחזרת המפסק למצב עקיבה:



איור 6 : המרה אחרונה והחזרת המפסק למצב עקיבה

אחרי שהבקרה הלוגית שמה '1' בביט הרביעי ב Test Value ה DAC מוציא מתח של $0.75V$ לכניסה ההופכת. היות והמתח בכניסת ה $+ (0.7v)$ קטן מהמתח בכניסת ה $-$ המשווה מוציא 0 לביט הרביעי ב Result מעביר את המפסק בחזרה למצב עקיבה (כדי לטעון את הקבל למתח הנוכחי של האות) ומודיע על סיום המרה.

7.2 מתח ייחוס

כל ממירי ה SAR ADC צריכים מתח ייחוס עבור ה DAC . מתח הייחוס קובע את מתח הכניסה שיגרום לקריאה מקסימלית (Full Scale) וישפיע על פונקציית המעבר/תגובה של ה ADC (יוסבר בהמשך). במיקרו בקרים של Silicon Labs ניתן לבחור את מתח הייחוס מבין מתח חיצוני או מתח פנימי או מתח ספק הכוח.

7.3 SAR CLOCK - פולסי השעון ל SAR

תדר השעון שמגיע ל SAR מתזמן את לולאות ההמרה ההשוואה המתבצעות וקובע באופן מעשי את זמן ההמרה , אבל לא את קצב הדגימה. קצב הדגימה הוא סכום של זמן העקיבה וזמן ההמרה. יש לבחור את תדר השעון ל SAR קרוב למקסימום הנתון בדף הנתונים ולא לעבור אותו.

7.4 איך להתחיל המרה ?

במיקרו בקרים של Silicon Labs אפשר להתחיל המרה מ 3 מקורות ראשיים :

- **כתיבה לביט "התחל המרה" הנקרא ADBUSY** . מצב זה נקרא אופן על פי דרישה On-demand כי ההמרה תתחיל מתי שאנחנו רוצים אותה. שימוש לאופן זה הוא בקריאה של מתח DC שצריך להמיר מידי פעם . דוגמאות אופייניות הן מדידת של

טמפרטורה או של מתח ספק כוח המתבצעות מידי פעם. כאשר נרצה לדגום מתחי חילופין נקבל ריצודים - jitter - קלים בגלל זמני הפקודה.

- **גלישות של טיימרים** שנמצאים במיקרו. הטיימרים במיקרו מספקים פולס "התחל המרה" כל זמן קבוע ומדויק. מצב זה שימושי לאותות חילופין - AC – או במצב הנקרא interrupt-driven system – מערכת מופעלת/מונעת פסיקות.
- **כניסה חיצונית הנקראת conversion-start (CNVSTR) - התחל המרה.** הכניסה החיצונית מאפשרת דגימות אסינכרוניות למערכת השעון של המיקרו. מצב זה שימושי כאשר רוצים דגימה המסונכרנת עם אירוע חיצוני.

7.5 - Right and Left Justification - יישור לימין ולשמאל.

ב ADC שיש לו יותר מ 8 ביטים מתקבלת תוצאת הקריאה בשני רגיסטרים. נניח שלרגיסטר קוראים ADC0 והתוצאה מתקבלת בשני הרגיסטרים ADC0L ADC0H. כל אחד מהם של 8 ביטים. ניתן דוגמה שקיבלנו קריאה של 26CH כלומר 1001101100 בינארי. הנתון נמצא ברגיסטרים ADC0H ADC0L בצורה הבאה:

ADC0H								ADC0L							
D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	0	0	0	1	0	0	1	1	0	1	1	0	0

כך ש $ADC0L = 01101100 = 6CH$ ו $ADC0H = 00000010 = 02H$. מצב זה שבו 6 הביטים מ D10 ועד D15 שווים 0 נראה לנו ברור והגיוני ולמעשה התוצאה בת 10 הביטים שהתקבלה מה ADC תהיה:

$$ADC0H = 000000D9D8 \quad ADC0L = D7D6D5D4D3D2D1D0$$

מצב זה נקרא **Right Justification** – יישור ימין.

קיים מצב שנקרא **Left Justification** – יישור שמאלה. במצב זה מזיזים את הנתונים של 2 הרגיסטרים 6 פעמים שמאלה כך שנקבל את התוצאה הבאה:

D9	D8	D7	D6	D5	D4	D3	D2	D1	D0	0	0	0	0	0	0
1	0	0	1	1	0	1	1	0	0	0	0	0	0	0	0

$$ADC0H = D9D8D7D6D5D4D3D2 \quad ADC0L = D1D0000000$$

או בדוגמה המספרית שלנו נקבל ב $ADC0L = 00000000 = 00H$ ו $ADC0H = 10001011 = 9BH$.

מדוע צריך את 2 האפשרויות של היישור ?

האפשרות הראשונה של Right Justification היא הגיונית וברורה. קיבלנו תוצאה מדויקת של 10 ביטים. הייתרון של יישור שמאלה הוא שניתן לקחת רק את הביט המשמעותי ביותר של הרגיסטר ונקבל דיוק רק של 8 ביטים כי 2 הביטים הנמוכים הם פחות משמעותיים. אפשרות זאת יכולה להיות שימושית במקרים הבאים:

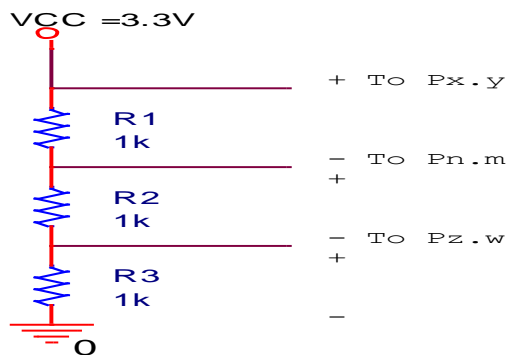
א. אם הדיוק איננו משמעותי.

ב. אם יש אילוץ של נפח זיכרון RAM שבו רוצים לשמור את הדגימות ואז כל קריאה היא של בית אחד ולא שני בתים וכך הכפלנו את נפח האחסון ללא פגיעה משמעותית בדיוק.

ג. כאשר מהירות היא נושא חשוב כי אז יש לקרוא את ADC0H בלבד ולשמור עליו בזיכרון ולא צריך לקרוא גם את ADC0L ושוב לשמור אותו בזיכרון, דבר שהיה מקטין את מהירות ריצת התוכנית וכמות הדגימות הנשמרת.

7.6 – מדידת מתח הפרשית ומדידה יחסית לאדמה - Single Ended .

בכניסות אל ה-ADC ישנם 2 מרבבים אנלוגיים. האחד מכניס לממיר את המתח החיובי והשני מכניס את המתח השלילי (הוא לא חייב להיות שלילי אלא הוא מתח שלילי יחסית לחיובי). בעזרת האיור הבא נסביר את ההבדל בין המרת מתח הפרשית והמרת מתח יחסית לאדמה הנקראת Single Ended. באיור מחובר מקור מתח של 3.3 וולט ל 3 נגדים שווים. על כל נגד נופל מתח של 1.1 וולט.



איור 7: המרת מתח הפרשית והמרה יחסית לאדמה.

הנקודות באיור הנקראות $x n z$ הם אחד הפורטים של המיקרו (מפורט מספר 0 ועד 4).

הנקודות $y m w$ מציינים את אחד ההדקים של הפורט (בין 0 ל 7).

אם נכניס ל $Ain+$ את המתח בהדק $Pn.m$ ולכניסה $Ain-$ את ההדק $Pz.w$ המדידה היא הפרשית ונקבל ערך דיגיטלי של המתח 1.1 וולט. המתח בנקודה $Pn.m$ חיובי ביחס ל $Pz.w$ וניתן גם לומר הפוך שהמתח בנקודה $Pz.w$ שלילי יחסית ל $Pn.m$. אם נכניס ב $Ain+$ את המתח בהדק $Pn.m$ ואילו בהדק $Ain-$ נכניס אדמה אז מצב זה נקרא **Single Ended** ונקבל קריאה דיגיטלית המתאימה למתח של 2.2 וולט.

7.6.1 מצב של מדידה Single-ended :

מתח הכניסה נכנס ל $Ain+$ ו $Ain-$ מחובר לאדמה האנלוגית AGND. $ADC Input = Ain+ - AGND$

מתח הכניסה המקסימלי הוא : $ADC Input = (Ain+ - AGND) = VREF$

מתח הכניסה המינימלי הוא : $ADC Input = (Ain+ - AGND) = 0 V$

במצב single-ended עבור ADC של 10 ביט הערך המקסימלי הוא 0X03FF.

בטבלה הבאה נציג את הקריאות של ה-ADC עבור מתחי כניסה שונים ועבור יישור ימני ויישור שמאלי.

RIGHT JUSTIFY**LEFT JUSTIFY**

VIN מתח כניסה	ADC0H	ADC0L	ADC0H	ADC0L
VREF*1023/1024	00000011	11111111	11111111	11000000
VREF*768/1024	00000011	00000000	11000000	00000000
VREF*512/1024	00000010	00000000	10000000	00000000
VREF*256/1024	00000001	00000000	01000000	00000000
VREF*128/1024	00000000	10000000	00100000	00000000
VREF*64/1024	00000000	01000000	00010000	00000000
0	00000000	00000000	00000000	00000000

טבלה 1

7.6.2 מצב של מדידה הפרשית :

המתח לכניסת ה ADC : $ADC\ Input = AIN+ - AIN-$. במקרה הזה ההפרש יכול להיות חיובי או שלילי.

לדוגמה : עבור $AIN+ = 1V$ ו $AIN- = 2V$ נקבל מתח כניסה הפרשי של $-1V$.

הפרש המתח בין כניסות ה $AIN+$ וה $AIN-$ הוא $\pm VREF$.

ההפרש המקסימלי : $ADC\ Input = (AIN+ - AIN-) = VREF$

הפרש המתח המינימלי בין הכניסות : $ADC\ Input = (AIN+ - AIN-) = -VREF$

בערך המקסימלי החיובי האפשרי הוא : $0X01FF$. שהוא 511 . הערך השלילי האפשרי הוא בשיטת המשלים ל 2 של מספר זה כלומר $0XFF00$.

נתאר בטבלה קריאות של ה ADC עבור מדידה הפרשית :

RIGHT JUSTIFY**LEFT JUSTIFY**

VIN מתח כניסה	ADC0H	ADC0L	ADC0H	ADC0L
VREF*511/512	00000001	11111111	01111111	11000000
VREF*256/512	00000001	00000000	01000000	00000000
VREF*128/512	00000000	10000000	00100000	00000000
VREF*64/512	00000000	01000000	00010000	00000000
0	00000000	00000000	00000000	00000000
-VREF	11111110	00000000	10000000	00000000
-VREF*511/512	11111110	00000001	10000000	10000000
-VREF*256/512	11111111	00000000	11000000	00000000
-VREF*128/512	11111111	10000000	11100000	00000000
-VREF*64/512	11111111	11000000	11110000	00000000

טבלה 2

7.7 חלון השוואה – חלון משווה - Window Comparator (לא נמצא בתוכנית הלימודים של כיתה יג)

במצב עבודה זה המשתמש קובע 2 ערכים המשמשים כחלון להשוואה. כאשר מתבצעת המרה ב ADC בודקים האם תוצאת ההמרה נמצאת בתוך החלון, כלומר בין שני הערכים שקבענו. אם כן - נוצרת בקשת פסיקה וביט המשמש כדגל מקבל '1'. פסיקה זו איננה קשורה לפסיקת סיום ההמרה של ה ADC.

קיים רגיסטר הנקרא "less-than" – "קטן מ..." ועוד רגיסטר "greater than" – "גדול מ..." לכל אחד מהרגיסטרים נותנים ערך. אם תוצאת ההמרה נמצאת בין הערכים שבשני הרגיסטרים מקבלים בקשה לפסיקת חלון ADC. ברכיבים במשפחת SILABS יש להם ADC של 10 או 12 ביטים קיימים 2 רגיסטרים לכל מתח סף - TRESH HOLD שבחלון. הרגיסטרים ייקראו **ADC0LTH** ו**ADC0LTL** שהם קיצור של **ADC0 Less-Than** (8 הביטים הנמוכים L והגבוהים H) ו**ADC0GTH** ו**ADC0GTL** שהם קיצור של **ADC0 Greater-Than** (8 הביטים הגבוהים והנמוכים).

באיור הבא רואים דוגמה שבה נותנים לרגיסטר less than ערך גבוה מזה שנותנים לרגיסטר greater than. כאשר תוצאת ההמרה בין שני ערכים אלו תיווצר בקשת פסיקת חלון.

כניסת ה ADC	יציאת ה ADC	
VREF-1LSB	0x0FFF	תחום ללא פסיקת חלון משווה
	0x0201	
	0x0200	ADC0LTH:ADC0LTL TRESHOLD (less-than)=0X200
	0x01FF	
החלון בין 2 הערכים	0x0101	עבור כל ערך בתוך החלון נוצרת בקשת פסיקת חלון משווה. מהערך 0x101 ועד הערך 0x1FF
	0x0100	
	0x00FF	ADC0GTH:ADC0GTL TRESHOLD greater-than = 0X100
	0x0000	
0	0x0000	תחום ללא פסיקת חלון משווה

איור 8 : חלון השוואה שבו לרגיסטר less than ערך גבוה מ greater than.

באיור רואים שרגיסטר less-than ("קטן מ...") קיבל ערך - 0X01FF - גבוה מרגיסטר greater-than - 0x0100. במקרה הזה עבור כל קריאה שיוציא הממיר שהיא בין 2 ערכים אלו נקבל פסיקה.

בדוגמה שבאיור הבא תתאר מצב שבו רגיסטר greater-than מקבל ערך גבוה מזה שברגיסטר less-than. במקרה הזה נקבל בקשת פסיקה עבור כל ערך מעל ערך הסף ברגיסטר greater than או מתחת לערך הסף ברגיסטר less-than.

כניסת ה ADC	יציאת ה ADC	
VREF-1LSB	0x0FFF	<div> <div>החלון בו נוצרת בקשת פסיקת חלון משווה.</div> <div>כל ערך מעל 0x200 ייצור פסיקת חלון משווה</div> </div>
Range of Interest	0x0201	
	0x0200	<div> <div>ADC0GTH:ADC0GTL</div> <div>TRESHOLD greater-than 0X200</div> </div>
	0x01FF	
	0x0101	<div> <div>בכל התחום הזה לא נוצרת בקשת פסיקה</div> </div>
	0x0100	
		<div> <div>ADC0LTH:ADC0LTL</div> <div>TRESHOLD (less-than) 0X100</div> </div>
Range of Interest	0x00FF	<div> <div>החלון בו נוצרת בקשת פסיקה חלון משווה.</div> <div>כל ערך מתחת 0x100 ייצור פסיקת חלון משווה</div> </div>
0	0x0000	

איור 9 : רגיסטר greater-than מקבל ערך גבוה מזה שברגיסטר less-than

7.8 מגבר בר תכנות (לא קיים ב C8051F380 !! ולא בתוכנית הלימודים) .

הרבה מיקרו בקרים מכילים ב ADC שלהם מגבר בר תכנות, כלומר מגבר שניתן לקבוע את ההגברה שלו. מגבר זה משמש להגדלת התחום הדינמי כאשר נכנס אות עם מתח נמוך בכניסה. בכניסת ה ADC יש מגבר שניתן לקבוע את ההגברה שלו. המגבר משנה את קנה המידה של היציאה כך שהוא מגביר אות חלש ואז ניתן לדייק יותר בקריאת התוצאה. נכון שבמקרה כזה גם הרעש גדל פי 4 אבל ניתן להתגבר על הרעש בעזרת דגימות רבות יותר ומיצוע (ממוצע) – oversampling and averaging, נושא שלא במסגרת הלימודים כאן.

7.9 מגבר הפרש מתח גבוה HVDA - High Voltage Difference Amplifier (לא קיים ב C8051F380 !!)

בחלק מ ADC של SILABS יש מגבר שהכניסות שלו יכולות לקבל מתחים של $\pm 60V$. ב ADC עם מגבר כזה ניתן למדוד אותות עם הפרשי מתח של $\pm 30V$. במגבר יש אפשרות להנחית את המתח הנכנס בעזרת מגברים שניתן לקבוע בתוכנה את ההנחתה שלהם. בצורה כזו מגיע ל ADC אות ברמות נתח מותרות.

7.10 מאפייני ADC

נסביר מספר מאפיינים של ממיר ADC אם כי בחלק מהם ניתן הגדרה בלבד ללא העמקה.

7.10.1 הפרדה – הבחנה – רזולוציה - RESOLUTION

הרזולוציה מציינת את מספר הערכים הדיסקרטיים שיכול הממיר ליצור. הערכים מיוצגים בצורה בינארית, כך שבדרך כלל מתארים אותה בביטים. לדוגמא: ממיר עם רזולוציה של 4 ביט הוא ממיר שנותן 16 רמות מתח אפשריות מ 0000 ועד 1111, סה"כ 2^4 . עבור ממיר של 8 ביט נקבל $2^8 = 256$ רמות מתח שונות, מ 00000000 עד 11111111 שיכולות להיות מ 0 עד 255 אם עובדים עם מספרים לא מסומנים או מ מינוס 128 עד פלוס 127 אם עובדים עם מספרים מסומנים). עבור ממיר בן 10 ביט הרזולוציה היא $2^{10} = 1024$ רמות מתח שונות. ניתן לתאר רזולוציה גם בצורה חשמלית בוולטים. לשינוי המינימאלי של המתח בכניסת הממיר הנדרש להבטיח שינוי במספר הבינארי ביציאה נקרא מתח ה LSB (Least Significant Bit). הרזולוציה של ADC שווה למתח ה LSB. ניתן לומר שמתח הרזולוציה של ממיר ADC למתח הייחוס לחלק במספר האפשרויות של הממיר:

$$\text{RESolution} = V_{\text{ref}} / 2^N$$

כאשר N הוא מספר הביטים של הממיר. V_{ref} הוא מתח הייחוס.

דוגמא: עבור ממיר ADC של 8 ביט עם מתח ייחוס של 5 וולט נקבל רזולוציה:

$$\text{RESolution} = 5 / 2^8 = 0.01953125\text{v}$$

ועבור ממיר של 10 ביט עם מתח ייחוס של 3.3 וולט כמו במיקרו C8051F380 הרזולוציה תהיה:

$$\text{RESolution} = 3.3 / 2^{10} = 0.00322265625\text{v} = 3.22265625\text{mV}$$

באופן מעשי הרזולוציה של הממיר מוגבלת על ידי יחס האות לרעש של האות הנבדק. אם הרעש המתלווה לאות גדול ממתח הרזולוציה, ברור שנקבל שגיאה בקריאה. מתח היציאה מהממיר – המספר הדיגיטאלי שנקבל – הוא לפי הנוסחה:

$$V_{\text{out}} = \text{ADCdata} = V_{\text{in}} / \text{RESolution} = V_{\text{in}} / (V_{\text{ref}}/2^n)$$

7.10.1.1 דוגמאות

דוגמה 1: נתון חיישן טמפרטורה המוציא מתח של 0.25 וולט בטמפרטורה של 25 מעלות צלסיוס. מהי הרזולוציה ומהו המתח שנקבל מממיר של 10 ביט עבור מתח $V_{\text{ref}} = 5\text{v}$?

$$\text{RESolution} = 5 / 2^{10} = 0.0048828125\text{v} \approx 4.88\text{mV}$$

$$\text{ADCdata} = 0.25 / (5/2^{10}) = 0.25 / 0.0048828125 = 51.2 \approx 51 = 33\text{H} = 00110011\text{B}$$

דוגמה 2 : Vref הוא 3.3v והמתח Vin הנכנס לממיר של 10 ביטים הוא 2 וולט . מהי הרזולוציה ומהי קריאת הממיר ?

$$REsolution = 3.3 / 2^{10} = 0.00322265625 \approx 3.22\text{mv}$$

$$ADCdata = 2 / (3.3 / 2^{10}) = 620.606 \approx 621 = 26DH = 10\ 0110\ 1101\ B$$

דוגמה 3 : בממיר שבדוגמה הקודמת – מהו מתח הכניסה המקסימלי שניתן להכניס בכניסה ?

המספר המקסימלי שניתן לקבל מממיר 10 ביטים הוא $2^n - 1$, כלומר 1023.

$$Vin_{MAX} = 1023 * REsolution = 1023 * 3.22 = 3.294\text{v}$$

7.10.2 סוג התגובה - RESPONSE TYPE

רוב הממירים הם ליניאריים, מושג שאומר שקיים יחס ליניארי בין תחום ערכי מתח היציאה ומתח הכניסה. ממירים מוקדמים היו בעלי תגובה לוגריתמית. בתוך הממיר הייתה רזולוציה גבוהה של 12 או 16 ביט ויציאת הממיר הייתה "ממופה" – מתוקנת – לקוד של 8 ביט.

7.10.3 דיוק - ACCURACY

לממיר ADC מספר מקורות שגיאה : א. שגיאת קוואנטיזציה. ב. שגיאת אי ליניאריות. ג. ריצוד השעון. שגיאות אלו נמדדות ביחידה שנקראת - least significant bit (LSB), שגיאת LSB (הביט הפחות משמעותי). בממיר של 8 ביט השגיאה היא 1/256 או 0.39 %.

7.10.4 שגיאת קוואנטיזציה - Quantization error

ההפרש בין האות המקורי והערך הספרתי שניתן לו. בשרטוט בעמוד קודם רואים שהערך שנתן הממיר לאות הוא 1010 ואות הכניסה לא בדיוק בערך הזה. השגיאה נובעת מהערכים הסופיים שיש לרזולוציה של הייצוג הספרתי של האות ולא ניתן להימנע ממנה. שגיאת קוואנטיזציה היא בין 0 לחצי LSB.

7.10.5 שגיאת אי ליניאריות - Non-linearity

שגיאה זו נובעת מאי שלמות פיזית של הממיר שגורמת ליציאה לא להיות ליניארית לגמרי.

7.10.6 דחיית ספק הכוח PSR-Power Supply Rejection

נתון דחיית ספק הכוח (PSR) מראה כיצד אדוות ספק הכוח מצומדת לכניסת ה ADC - באופן שמשפיע על המוצאים הדיגיטליים של הממיר. עם PSR מוגבל, רעש על קו ספק הכוח יונחת רק ל-30 עד 50 דציבל מתחת לרמת הכניסה. הגודל PSR חשוב לקביעת רמת הסינון ונטילת צימוד של ספק הכוח. הדבר חשוב בסביבות של רעש גבוה, כמו ביישומים רפואיים או תעשייתיים.

7.10.7 דחיית אות באופן משותף

נתון ה Common-Mode Rejection (CMR) - מודד את האות הדיפרנציאלי המושרה בנוכחות של אות באופן משותף (common-mode). התקני ADC רבים משתמשים בכניסות דיפרנציאליות כדי לספק חסינות גבוהה מפני אותות באופן משותף, מאחר שמבני כניסה דיפרנציאליים דוחים באופן טבעי תוצרי עיוות בסדר זוגי.

7.10.8 קצב עליית השעון

קצב עליית השעון (clock slew rate) הוא קצב העלייה המינימאלי הנדרש כדי להשיג את הביצועים שהוגדרו. למרבית הממירים יש מספיק שבח על חוצץ השעון כדי להבטיח שרגע הדגימה יוגדר היטב, אבל רעש עודף יופיע אם קצב העלייה איטי מספיק כדי לייצר דרגה גבוהה של אי ודאות לגבי רגע הדגימה. אם מוגדר קצב עלייה מינימאלי בכניסה, המשתמשים צריכים לעמוד בדרישה זו כדי להבטיח את ביצועי הרעש שהוגדרו.

7.10.9 ריצוד מיפתח

ריצוד מיפתח (aperture jitter) הוא אי-הוודאות הפנימית של השעון ל-ADC-ביצועי הרעש של ה-ADC - מוגבלים על-ידי ריצוד השעון, הן הפנימי והן החיצוני. בגיליון נתונים טיפוס, ריצוד מיפתח מתייחס לממיר בלבד. ריצוד מיפתח חיצוני מסתכם בצורת RMS עם ריצוד מיפתח פנימי. עבור יישומי תדר-נמוך, הריצוד עשוי להיות טפל, אבל ככל שהתדר האנלוגי גדל, הרעש שמקורו בריצוד הופך לדאגה גוברת. אי שימוש בשעון הולם יוביל לביצועים ירודים מן הצפוי. בנוסף לרעש מוגבר מריצוד השעון, יופיעו קווים ספקטראליים באות האנלוגי, אשר אינם קשורים מבחינה הרמונית לשעון, כעיוות על המוצא המסופרת. לפיכך, אות השעון אמור להיות עם הטוהר הספקטראלי הגבוה ביותר האפשרי.

7.10.10 השהיית מיפתח

השהיית מיפתח (aperture delay) היא פרק הזמן שבין היישום של אות הדגימה לבין הרגע שבו אות הכניסה נדגם בפועל. זמן זה – באופן טיפוס ננו-שנייה או פחות – עשוי להיות חיובי, שלילי או אפילו אפס. השהיית המיפתח אינה חשובה בדרך כלל, למעט מקרים בהם ידיעת רגע הדגימה המדויק משמעותית.

7.10.11 זמן המרה ושיהוי המרה

זמן המרה (conversion time) ושיהוי המרה (conversion latency) הם שני מפרטים הקשורים ביניהם באופן הדוק. זמן המרה מתייחס בדרך-כלל לממירי SAR (successive-approximation) בהם קצב שעון מהיר משמש לעיבוד אות הכניסה, אשר מופיע על המוצא הרבה יותר מאוחר מאשר פקודת ההמרה, אבל קודם לפקודת ההמרה הבאה. הזמן שבין פקודת ההמרה לבין השלמת ההמרה הוא זמן ההמרה.

שיהוי המרה מתייחס בדרך-כלל לממירים מסוג צינור (pipelined). השיהוי, הוא מדד של מספר הצינורות (שלבים דיגיטליים פנימיים) אשר משמשים לייצור הפלט הדיגיטלי, מוזכר לרוב במונחים של השהיות צינור. ניתן לחשב זמן המרה בפועל על-ידי הכפלת מספר זה בפרק זמן הדגימה המשמש ביישום.

7.10.12 זמן התעוררות

כדי לשמר הספק ביישומים המודעים להספק, מתנתק ההתקן בדרך-כלל במהלך פרקי זמן של חוסר שימוש יחסי. אף כי צעד זה חוסך הספק ניכר, כמות סופית של זמן נדרשת לייחוסים פנימיים כדי להתייצב ולפונקציונאליות של השעון הפנימי לשוב ולתפקד כאשר ההתקן נדלק שוב. במהלך פרק זמן זה, נתוני ההמרה הנוצרים לא יעמדו במפרטים.

7.10.13 עומס מוצא

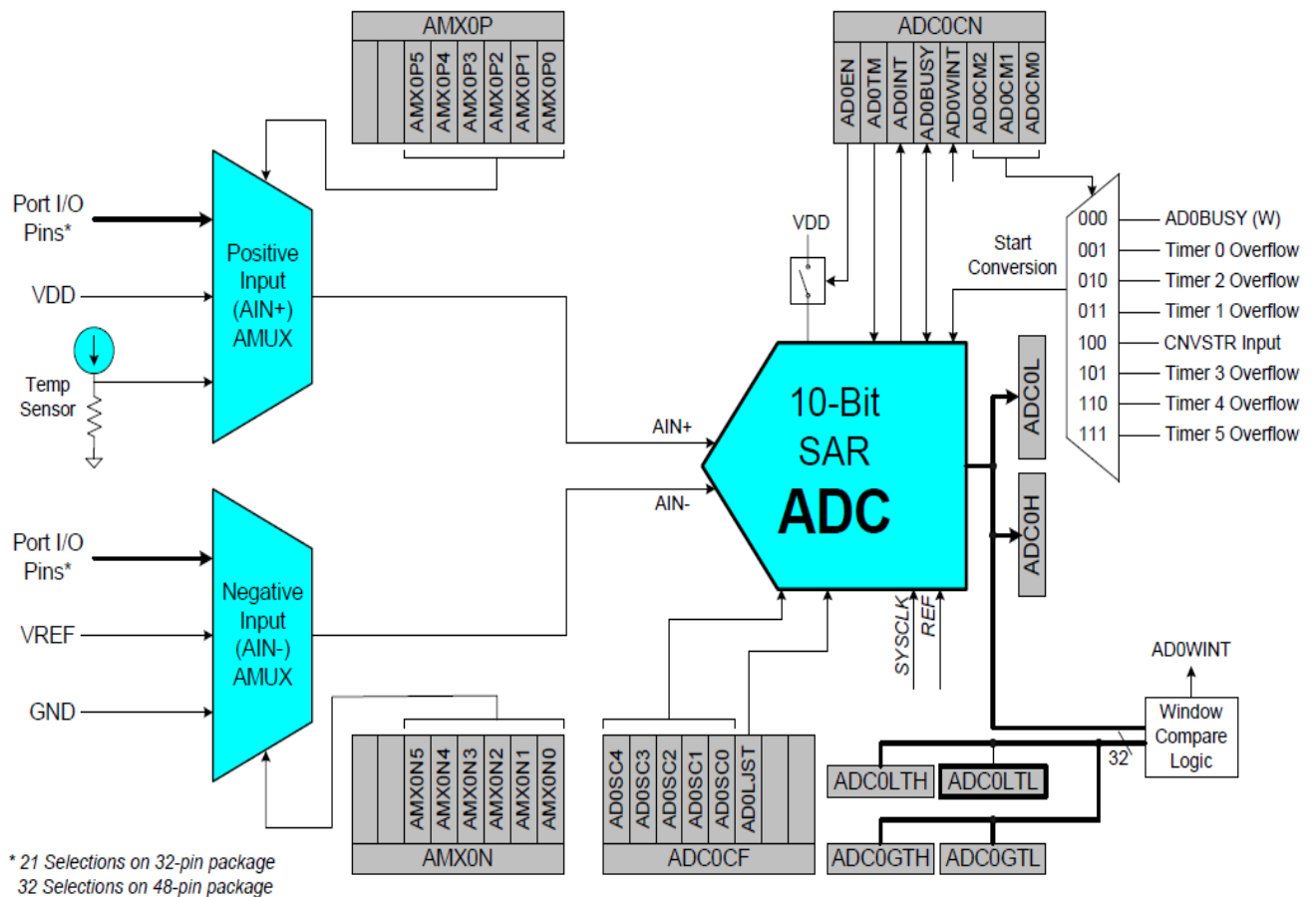
כמו כל התקני הפלט הדיגיטליים, התקני ADC – בעיקר התקנים עם יציאת CMOS מגדירים את יכולת דחף המוצא. אף כי חשוב לדעת נתון זה מסיבות של אמינות, הביצועים האופטימאליים יתרחשו בדרך-כלל בפחות ממלוא יכולת הדחף. ביישומים הדורשים ביצועים גבוהים, חשוב למזער את עומס המוצא ולספק נטילת צימוד הולמת ועריכה אופטימאלית כדי למזער את נפילת המתח על ספק הכוח. כדי להימנע מבעיות אלה, מספקים ממירים רבים מוצאי LVDS. מאחר ש LVDS – סימטרי, זרמי מיתוג מצטמצמים והביצועים הכוללים משתפרים. אם יציאות LVDS זמינות, צריך לעשות בהן שימוש כדי להבטיח את הביצועים הטובים ביותר.

7.11 ממיר ה ADC ברכיב C8051F380 - ADC0

ממיר ה ADC ברכיב נקרא ADC0 ויכול לבצע דגימות בקצב של עד 500ksps (Kilo Samples Per Second) – קילו דגימות בשנייה). הממיר בן 10 ביטים מסוג SAR - Successive Approximation Register (שערוך מוצלח) עם מערכת עקיבה והחזקה - Track and Hold - וגלאי חלון מתוכנת (המושגים האלו יוסברו בהמשך). ניתן לקבוע בעזרת תוכנה את קצב ההמרה (עד למקסימום של 500 אלף דגימות בשנייה), לבחור האם להכניס ל ADC מתח אנאלוגי יחסית לאדמה או מתח צף (הפלוס והמינוס לא מיוחסים לאדמה), לבחור בעזרת מעגל AMUX (מרבב אנאלוגי) מאיזו כניסה שלו (ניתן לחבר ל MUX עד 32 מקורות מתח שונים) ייכנס המתח להמרה וכמו כן ניתן לקבוע את מתח ה Vref שהממיר יעבוד בו. האיור הבא מתאר את מערכת ה ADC של הרכיב.

- **הממיר - Bit SAR ADC - 10** נמצא במרכז האיור מימין. מצד שמאל שלו יש את כניסות המתח אליו. AIN+ ו AIN-.
- **ADC0H ADC0L** – החלק הגבוה והחלק הנמוך של ה ADC מימין לממיר ה ADC. אלו 2 הרגיסטרים שבהם נמצא הערך הדיגיטאלי של המתח האנאלוגי שנכנס לממיר. היות ויש לממיר 10 ביטים אז בחלק הנמוך – ADC0L יהיו את 8 הביטים הנמוכים של הערך ואילו בחלק הגבוה ADC0H יהיו את שני הביטים הגבוהים של הערך (ישנה אפשרות נוספת של ישור שמאלה שמוזכרת בסעיפים קודמים).
- **AMUX 2 – מרבבים אנלוגיים**. כל מרבב הוא מ $1 \rightarrow 32$. לכל AMUX יש 32 כניסות ויציאה אחת. ה AMUX העליון - Positive Input (AIN+) AMUX - מעביר לממיר את אחת מ 32 הכניסות ויציאה זו תהיה הפלוס של מתח הכניסה (AIN+) וה AMUX התחתון - Negative Input (AIN-) AMUX - מעביר ל ADC את אחת מ 32 הכניסות המתחברות אליו. היציאה שלו תהיה המתח השלילי של מתח הכניסה (AIN-). יכול להיות מצב שבו AIN+ יהיה קטן מ AIN- ואז נקבל ערך שלילי ברגיסטרים

- הממיר כולל את היחידות הבאות: **רגיסטרים AMX0P** - שנמצא מעל Positive Input AIN+ AMUX שולט על ה AMUX החיובי וקובע איזו כניסה מתוך ה 32 תצא ותגיע ל AIN+ של הממיר. הרגיסטר AMX0N למטה משמאל פועל בצורה דומה לגבי AIN-.



איור 10 המבנה הפנימי של ה ADC .

- **רגיסטר ADC0CN** - ADC0 CoNtrol - בקרת ADC - שנמצא למעלה מימין הוא הרגיסטר בקרה של ה ADC ובעזרתו קובעים האם ה ADC יעבוד או לא , מאיפה תגיע פקודת "התחל המרה" Start Conversion (נסביר בהמשך), פסיקת חלון משווה ועוד.
- **רגיסטר ADC0CF** - ADC0 ConFigation - קביעת התצורה של ה ADC שנמצא למטה מימין ותפקידו לקבוע את קצב הדגימות והאם על תוצאת הקריאה (המספר הדיגיטאלי המתקבל) יבוצע יישור - Justification - ימינה או שמאלה (נסביר בהמשך) .
- **ADC0LTH ADC0LTL ו ADC0GTH ADC0GTL** הם 4 רגיסטרים לעבודה עם גלאי חלון שמוסבר בסעיפים קודמים על משווה החלון. בעזרת ADC0LTH ADC0LTL קובעים את ערך הסף של Less-Than ובעזרת הרגיסטרים ADC0GTH ADC0GTL קובעים את ערך הסף של Greater-Than .
- מצד ימין של הממיר למעלה נמצא מרבב. בעזרת מרבב זה קובעים מאיזה מקור יגיע האות "התחל המרה" - Start Conversion . המרבב נשלט על ידי 3 ביטים ברגיסטר ADC0CN

- בחלקו התחתון מקבל הממיר 2 כניסות של SYSCLK – שעון מערכת ו REF מתח ייחוס. ניתן לתכנת לשתי הכניסות האלו את תדר השעון ואת מתח הייחוס כפי שנסביר בהמשך.

7.12 הרגיסטרים של ADC0

נסביר את תפקידי הרגיסטרים הקשורים ל ADC ונתחיל עם 2 המרבבים האנלוגיים הנקראים AMUX0P ו AMUX0N בעזרתם בוחרים את הכניסה החיובית Ain+ והשלילית Ain- שיגיעו להמרה ב ADC .

7.12.1 AMUX0P – המרבב הבוחר את הכניסה החיובית AIN+

הכניסה החיובית AIN+ יכולה להיות הדק כלשהו של פורט או לחיישן טמפרטורה שנמצא בתוך האעיב או למתח הספק החיובי VDD . הכניסה השלילית AIN- יכולה להתחבר אל הדק כלשהו של פורט, או ל VREF או לאדמה. את ההדקים של פורט כלשהו שמשמשים ככניסה ל ADC יש לקבוע כהדקים אנאלוגיים ויש לדלג עליהם בקרוסבר . כדי לקבוע הדק כאנלוגי יש לשים '0' בביט המתאים ברגיסטר PnMDIN . כדי לאלץ את הקרוסבר לדלג על הדק של פורט יש לשים '1' בביט המתאים ברגיסטר PnSKIP .

נתחיל עם המרבב האנלוגי של הכניסה החיובית +VIN הנקרא AMUX0P . הוא נראה באיור הבא :
 2 הביטים הגבוהים אינם בשימוש. 6 הביטים הנמוכים מ 0 ועד 5 קובעים איזה הדק של פורט יתחבר אל הממיר .
 ישנן 2 עמודות עבור C8051F380 שיש בו זיווד/אריזה - PACKAGE - של 32 הדקים ועבור C8051F380 שיש לו זיווד/אריזה עם 48 הדקים. לדוגמה אם נרשום : AMUX0P=0x0 ; אז ברכיב עם ה 32 הדקים ייכנס לממיר המתח שנכניס ב P1.0 ואילו ברכיב עם 48 הדקים ייכנס לממיר המתח בהדק P2.0 .
 ברכיב עם הזיווד של 32 הדקים בחלק ממצבי הביטים לא מתחבר הדק כלשהו אל הממיר ונרשם שם Reserved .
 אם נרשום : AMUX0P=0x1E; אז ייכנס ל Ain+ של ה ADC מתח מחיישן הטמפרטורה הפנימי שנמצא ברכיב. במקרה הזה לא משנה אם הזיווד הוא של 32 הדקים או 48 הדקים.

SFR Definition 6.9. AMX0P: AMUX0 Positive Channel Select

Bit	7	6	5	4	3	2	1	0
Name	AMX0P[5:0]							
Type	R/W							
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xBB; SFR Page = All Pages

Bit	Name	Function					
7:6	Unused	Read = 00b; Write = don't care.					
5:0	AMX0P[5:0]	AMUX0 Positive Input Selection.					
		AMX0P	32-pin Packages	48-pin Packages	AMX0P	32-pin Packages	48-pin Packages
		000000:	P1.0	P2.0	010010:	P0.1	P0.4
		000001:	P1.1	P2.1	010011:	P0.4	P1.1
		000010:	P1.2	P2.2	010100:	P0.5	P1.2
		000011:	P1.3	P2.3	010101:	Reserved	P1.0
		000100:	P1.4	P2.5	010110:	Reserved	P1.3
		000101:	P1.5	P2.6	010111:	Reserved	P1.6
		000110:	P1.6	P3.0	011000:	Reserved	P1.7
		000111:	P1.7	P3.1	011001:	Reserved	P2.4
		001000:	P2.0	P3.4	011010:	Reserved	P2.7
		001001:	P2.1	P3.5	011011:	Reserved	P3.2
		001010:	P2.2	P3.7	011100:	Reserved	P3.3
		001011:	P2.3	P4.0	011101:	Reserved	P3.6
		001100:	P2.4	P4.3	011110:	Temp Sensor	Temp Sensor
		001101:	P2.5	P4.4	011111:	V _{DD}	V _{DD}
		001110:	P2.6	P4.5	100000:	Reserved	P4.1
		001111:	P2.7	P4.6	100001:	Reserved	P4.2
		010000:	P3.0	Reserved	100010:	Reserved	P4.7
		010001:	P0.0	P0.3	100011 - 111111:	Reserved	Reserved

איור 11 : AMX0P - בחירת הכניסה החיובית.

7.12.2 AMUX0N – המרבב הבוחר את הכניסה השלילית -AIN.

בעזרת המרבב הזה בוחרים איזו הדק של הרכיב יתחבר אל הכניסה השלילית של הממיר -Ain . ניתן גם לקבוע שבמקום שייכנס מתח מאחד ההדקים ייכנס להדק השלילי אדמה ואז המצב נקראה single ended והמתח הנכנס ב AIN+ נמדד יחסית לאדמה. גם כאן יש 2 עמודות עבור רכיבים עם זיווד של 32 הדקים ורכיבים עם 48 הדקים. אם נרשום ; AMUX0N = 0x1F אז המדידה תהיה יחסית לאדמה.

SFR Definition 6.10. AMX0N: AMUX0 Negative Channel Select

Bit	7	6	5	4	3	2	1	0
Name	AMX0N[5:0]							
Type	R	R	R/W					
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xBA; SFR Page = All Pages

Bit	Name	Function					
7:6	Unused	Read = 00b; Write = don't care.					
5:0	AMX0N[5:0]	AMUX0 Negative Input Selection.					
		AMX0N	32-pin Packages	48-pin Packages	AMX0N	32-pin Packages	48-pin Packages
		000000:	P1.0	P2.0	010010:	P0.1	P0.4
		000001:	P1.1	P2.1	010011:	P0.4	P1.1
		000010:	P1.2	P2.2	010100:	P0.5	P1.2
		000011:	P1.3	P2.3	010101:	Reserved	P1.0
		000100:	P1.4	P2.5	010110:	Reserved	P1.3
		000101:	P1.5	P2.6	010111:	Reserved	P1.6
		000110:	P1.6	P3.0	011000:	Reserved	P1.7
		000111:	P1.7	P3.1	011001:	Reserved	P2.4
		001000:	P2.0	P3.4	011010:	Reserved	P2.7
		001001:	P2.1	P3.5	011011:	Reserved	P3.2
		001010:	P2.2	P3.7	011100:	Reserved	P3.3
		001011:	P2.3	P4.0	011101:	Reserved	P3.6
		001100:	P2.4	P4.3	011110:	VREF	VREF
		001101:	P2.5	P4.4	011111:	GND (Single-Ended Measurement)	GND (Single-Ended Measurement)
		001110:	P2.6	P4.5	100000:	Reserved	P4.1
		001111:	P2.7	P4.6	100001:	Reserved	P4.2
		010000:	P3.0	Reserved	100010:	Reserved	P4.7
		010001:	P0.0	P0.3	100011 - 111111:	Reserved	Reserved

איור 12 : AMX0N - בחירת הכניסה השלילית.

7.12.3 - הרגיסטר ADC0CN - ADC0 CoNtrol - רגיסטר הבקרה של ADC0

בעזרת רגיסטר זה מבקרים על פעולת ADC0. הוא מתואר באיור הבא:

SFR Definition 6.4. ADC0CN: ADC0 Control

Bit	7	6	5	4	3	2	1	0
Name	AD0EN	AD0TM	AD0INT	AD0BUSY	AD0WINT	AD0CM[2:0]		
Type	R/W	R/W	R/W	R/W	R/W	R/W		
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xE8; SFR Page = All Pages; Bit-Addressable

Bit	Name	Function
7	AD0EN	ADC0 Enable Bit. 0: ADC0 Disabled. ADC0 is in low-power shutdown. 1: ADC0 Enabled. ADC0 is active and ready for data conversions.
6	AD0TM	ADC0 Track Mode Bit. 0: Normal Track Mode: When ADC0 is enabled, tracking is continuous unless a conversion is in progress. Conversion begins immediately on start-of-conversion event, as defined by AD0CM[2:0]. 1: Delayed Track Mode: When ADC0 is enabled, input is tracked when a conversion is not in progress. A start-of-conversion signal initiates three SAR clocks of additional tracking, and then begins the conversion. Note that there is not a tracking delay when CNVSTR is used (AD0CM[2:0] = 100).
5	AD0INT	ADC0 Conversion Complete Interrupt Flag. 0: ADC0 has not completed a data conversion since AD0INT was last cleared. 1: ADC0 has completed a data conversion.
4	AD0BUSY	ADC0 Busy Bit. Read: 0: ADC0 conversion is not in progress. 1: ADC0 conversion is in progress. Write: 0: No Effect. 1: Initiates ADC0 Conversion if AD0CM[2:0] = 000b
3	AD0WINT	ADC0 Window Compare Interrupt Flag. 0: ADC0 Window Comparison Data match has not occurred since this flag was last cleared. 1: ADC0 Window Comparison Data match has occurred.
2:0	AD0CM[2:0]	ADC0 Start of Conversion Mode Select. 000: ADC0 start-of-conversion source is write of 1 to AD0BUSY. 001: ADC0 start-of-conversion source is overflow of Timer 0. 010: ADC0 start-of-conversion source is overflow of Timer 2. 011: ADC0 start-of-conversion source is overflow of Timer 1. 100: ADC0 start-of-conversion source is rising edge of external CNVSTR. 101: ADC0 start-of-conversion source is overflow of Timer 3. 110: ADC0 start-of-conversion source is overflow of Timer 4. 111: ADC0 start-of-conversion source is overflow of Timer 5.

איור 13 : רגיסטר הבקרה ADC0CN

ביט **AD0EN** - ADC0 Enable - אפשרור ADC0. אם נשים 0 – לא מאפשרים את פעולת ADC0 והוא איננו פועל וצורך הספק נמוך – Low Power. אם נשים 1 ADC0 הממיר פועל ומוכן להמרה.

AD0TM - ADC0 Track Mode - הביט קובע את אופן העקיבה. אם נשים '0' - מצב עקיבה רגיל. העקיבה מתחילה מיד אחרי פקודת "התחל המרה" – start conversion – כפי שמוגדר בביטים AD0CM[2:0] שהם ביטים 0 עד 2 של הרגיסטר. אם נשים '1' – זהו מצב עקיבה מושהה. העקיבה מושהית עוד 3 פולסי SAR (כדי שהקבל יספיק להיטען) ורק אז מתחילה ההמרה. הסבר על מצב עקיבה הוסבר בסעיפים קודמים ויוסבר מיד כאשר נדבר על הביטים 0 עד 2 של הרגיסטר.

AD0INT - ADC0 INTerrupt - פסיקת ADC0 - כאשר מסתיימת המרה הביט עולה ל 1. יש לשים בו 0 בתוכנה לפני התחלת המרה חדשה. אם נעבוד עם שאילתה - **POLLING** נבדוק את הביט הזה כדי לדעת האם הסתיימה המרה.

AD0BUSY - ADC0 BUSY - כאשר יש בו 0 זה אומר שלא מתבצע תהליך המרה בממיר וכאשר יש בו 1 זה אומר שהממיר עסוק ומבצע המרה. קיימת אפשרות שהמשתמש ייקבע בתוכנה את פולס "התחל המרה". זה קורה כאשר עובדים עם הביטים 0 עד 2 של הרגיסטר - **ADC0CM[2:0]** - במצב 000. אם כותבים לביט הזה '1' זה גורם לממיר להתחיל המרה. בסיום ההמרה הביט יורד ל 0 כדי להודיע שהממיר לא עסוק ובנוסף הביט הקודם **AD0INT** עולה ל 1 ויש בקשת פסיקה.

AD0WINT - ADC0 Window compare INTerrupt flag - ביט זה מראה האם הייתה התאמה בחלון ההשוואה. אם יש בו 1 זה מראה שהייתה התאמה. אם יש בו 0 לא הייתה התאמה מאז שאיפסנו את הביט בתוכנה. כאשר יש בו 1 ורוצים לחפש התאמת חלון חדשה יש לאפס את הביט בתוכנה.

ADC0M[2:0] - 3 ביטים אלו קובעים את המקור ממנו תגיע פקודת "התחל המרה" - **start conversion**, כלומר מאיפה תגיע פקודת "התחל המרה" לפי 8 המצבים הבאים:

000 - כאשר כותבים לביט 4 - **AD0BUSY** ברגיסטר כאן (**ADC0CN**).

001 - גלישה של טיימר 0.

010 - גלישה של טיימר 2.

011 - גלישה של טיימר 1.

100 - עלייה (מעבר מ 0 ל 1) בהדק **CNVSTR**. ההדק P1.4 משמש כהדק "התחל המרה".

101 - גלישה של טיימר 3.

110 - גלישה של טיימר 4.

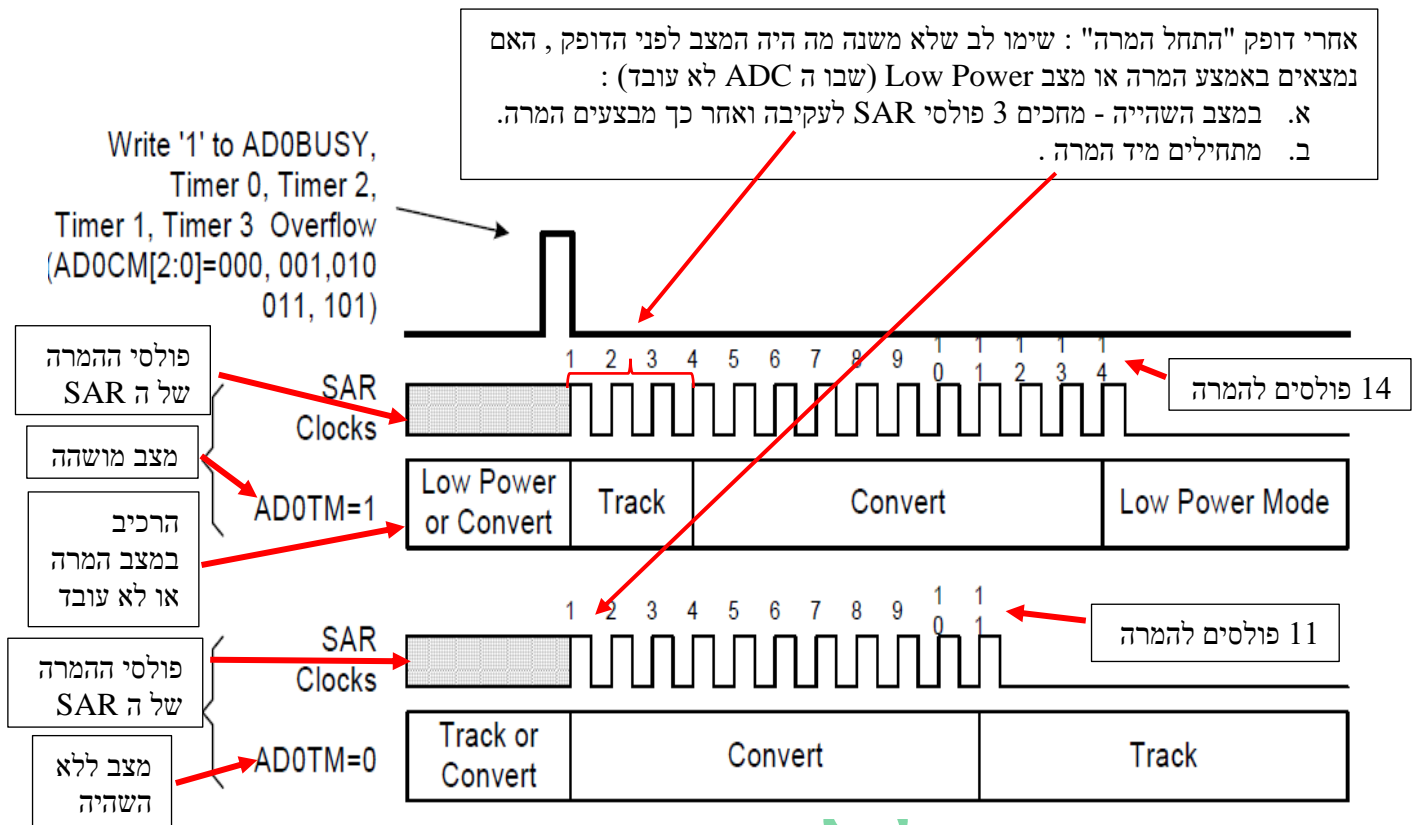
111 - גלישה של טיימר 5.

ניתן לראות שכל המצבים חוץ במצב 100 הם מצבים שפקודת "התחל המרה" הם ממקור פנימי. במצב 100 המקור הוא חיצוני והוא יגיע מאות שייכנס להדק P1.4 של הרכיב.

7.12.3.1 - תוספת לאופן עקיבה - **Tracking Mode** ואופן עקיבה מושהה.

הביט **AD0TM - ADC0 Track Mode** (ביט מספר 6 ב **ADC0CN**) קובע את אופן העקיבה והחזקה של הממיר. ברירת המחדל שלו היא **ADC0TM=0**, כלומר עקיבה מתמשכת חוץ מהמקרה שנמצאים בתהליך המרה. כאשר **AD0TM=1** עובדים באופן עקיבה והחזקה עם תצרוכת הספק נמוכה. במצב זה לכל התחלה של המרה קודמים 3 פולסי שעון **SAR** (אחרי פקודת "התחל המרה"). כאשר האות **CNVSTR** משמש התחלת המרה העקיבה באופן עקיבה בהספק נמוך, מתבצעת עקיבה רק כאשר **CNVSTR** בנמוך. ההמרה מתחילה כאשר יש עליה ב **CNVSTR**.

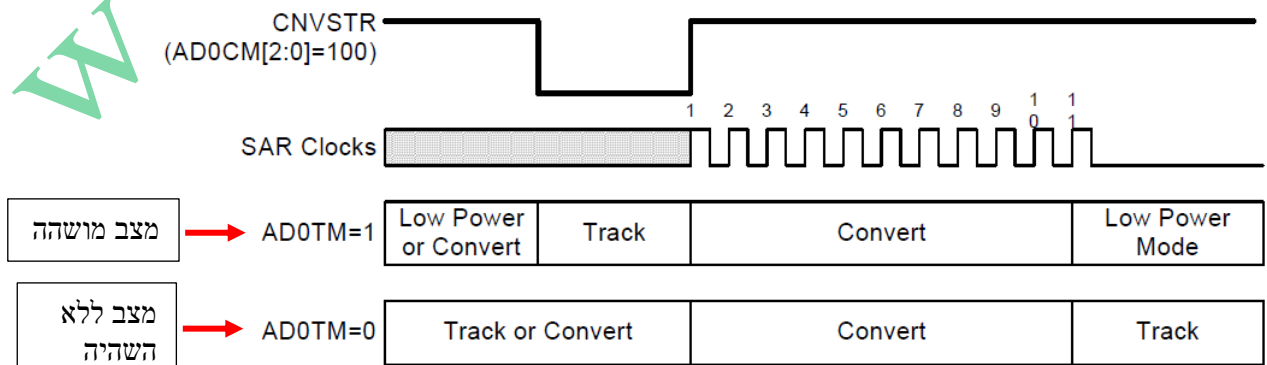
האיור הבא מתאר את ההבדלים בין 0 ו 1 בביט **ADC0TM**. המצב מתאים לכל המצבים של הביטים 0,1,2 **ADC0CM[2:0]** שהוא מצב שבו "התחל המרה" הוא על ידי מקור פנימי חוץ ומתחתיו מצב של 100 שמקור "התחל המרה" הוא חיצוני.



איור 14 : זמנים ב ADC0 עם מקור פנימי של "התחל המרה".

בחלק העליון רואים את דופק "התחל המרה". שתי צורות הגל מתחת לדופק "התחל המרה" (SAR Clocks ו AD0TM=1) הם עבור מצב מושהה. מיד עם כתיבת '1' לביט הממיר מתחיל בהמרה ומוריד את הביט ל 0. אם AD0TM=1 (מצב מושהה) מחכים 3 פולסי SAR ורק אז מתחילה ההמרה. סה"כ זמן ההמרה הם 14 מחזורי שעון SAR. צורות הגלים האחרונות מתארות מצב רגיל ללא השהייה. זהו המצב של AD0TM=0 שבו לא משנה אם היינו במצב המרה או עקיבה, מתחילה ההמרה חדשה מיד בסיום דופק "התחל המרה". זמן ההמרה הוא 11 מחזורי SAR.

האיור הבא מתאר מצב של "התחל המרה" על ידי מקור חיצוני בהדק CNVSTR :



איור 15 : זמנים ב ADC0 עבור מקור חיצוני של "התחל המרה".

מהאיור רואים שכאשר ההדק של המיקרו CNVSTR הוא ב 0 אז במצב עם השהיה מתבצעת קודם עקיבה ומיד עם העלייה של ההדק ל '1' מתחילה ההמרה. במצב ללא השהיה לא משנה אם היינו במצב עקיבה או המרה, העלייה של ההדק ל '1' מתחילה את ההמרה.

7.12.4 רגיסטר ADC0CF - ADC0 ConFfiguration - תצורה של ADC0

בעזרת רגיסטר זה קובעים את קצב פולסי השעון שיגיעו ל SAR ואת היישור (ימני או שמאלי).

האיור הבא מתאר את הרגיסטר:

Bit	7	6	5	4	3	2	1	0
Name	AD0SC[4:0]					AD0LJST	Reserved	
Type	R/W					R/W	R/W	
Reset	1	1	1	1	1	0	0	0

SFR Address = 0xBC; SFR Page = All Pages

Bit	Name	Function
7:3	AD0SC[4:0]	ADC0 SAR Conversion Clock Period Bits. SAR Conversion clock is derived from system clock by the following equation, where AD0SC refers to the 5-bit value held in bits AD0SC4–0. SAR Conversion clock requirements are given in the ADC specification table. $AD0SC = \frac{SYSCLK}{CLK_{SAR}} - 1$ Note: If the Memory Power Controller is enabled (MPCE = '1'), AD0SC must be set to at least "00001" for proper ADC operation.
2	AD0LJST	ADC0 Left Justify Select. 0: Data in ADC0H:ADC0L registers are right-justified. 1: Data in ADC0H:ADC0L registers are left-justified. Note: The AD0LJST bit is only valid for 10-bit mode (AD08BE = 0).
1:0	Reserved	Must Write 00b.

איור 16 - רגיסטר התצורה ADC0CF.

5 הביטים הגבוהים AD0SC[4:0] AD0 Sar conversion Clock period bits - קובעים את קצב פולסי השעון המגיעים ל ADC. הקצב נקבע לפי הנוסחה הבאה:

$$1. \quad AD0SC = \frac{SYSCLK}{CLK_{SAR}} - 1$$

AD0SC הם הערך ב 5 הביטים SD0SC[4:0]. CLK_{SAR} הוא תדר הפולסים ל ADC. $SYSCLK$ תדר שעון המערכת. כדי למצא את קצב פולסי השעון המגיעים ל ADC נעשה שינוי נושא בנוסחה נקבל:

$$2. \quad CLK_{SAR} = SYSCLK / (AD0SC + 1)$$

ביט מספר 2 AD0LJST - ADC0 Left Justify Select - קובע האם היישור יהיה שמאלי או ימני. במצב 0 הוא היישור הוא ימני במצב של 1 היישור הוא שמאלי.

ביטים 0 ו 1 לא בשימוש.

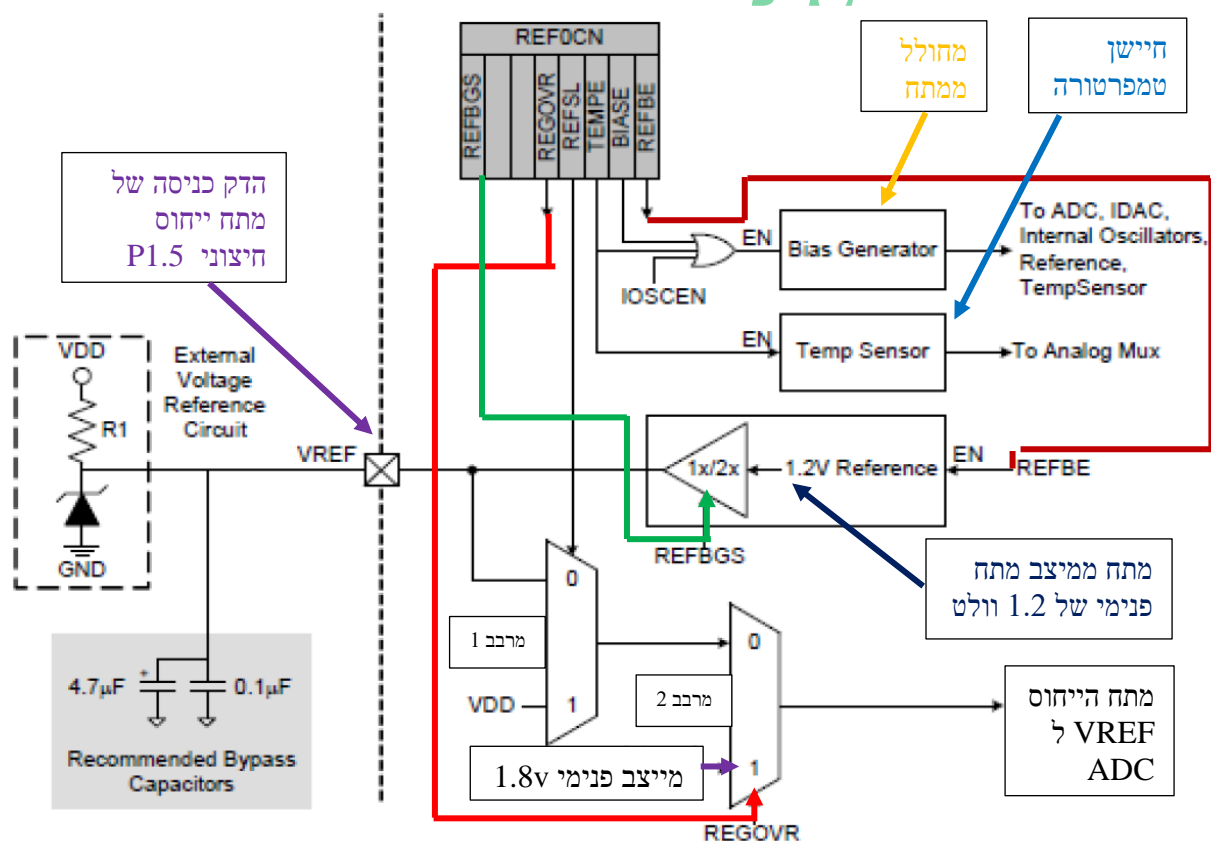
דוגמה : עבור תדר מערכת של 12MHZ. מהו הערך שיש לכתוב בביטים AD0SC עבור קצב דגימה של 500000 דגימות בשנייה. ניעזר במשוואה מספר 1 :

$$AD0SC = 12 \cdot 10^6 / 500000 - 1 = 23$$

זהו קצב הדגימה של הפולסים המגיעים ל ADC. יש לתכנן שפולס "התחל המרה" – Start Conversion יגיע בזמן קרוב ככל האפשר לקצב זה. אם פולס התחל המרה מגיע מאחד הטיימרים אז יש לדאוג שהטיימר יגלוש בקצב קרוב לקצב זה. ליתר דיוק יש לתכנן זמן גלישה קצת יותר מזמן של 11 פולסים שדרושים להמרה כדי שהקבל בכניסת ה ADC יספיק להיטען.

7.13 מתח הייחוס VREF ל ADC.

מתח הייחוס יכול להיבחר מבין מתח ייחוס חיצוני או מתח ה VDD של ספק הכוח או מייצב מתח פנימי. האיור הבא מתאר את מערכת בחירת VREF.



איור 17 : מערכת בחירת מתח הייחוס

במרכז למעלה נמצא רגיסטר REF0CN - REFeRence 0 CoNtrol – בקרת ייחוס - המבקר על מתח הייחוס שיגיע ל ADC, האם להפעיל חיישן טמפרטורה שנמצא במיקרו בקר ועוד. מתח הייחוס ל ADC יכול להגיע ממספר מקורות:

1. מתח ייחוס שהשתמש יכול להכניס בהדק חיצוני של המיקרו (באיור מצד שמאל באמצע המלבן עם ה X) שאליו מכניסים מתח ייחוס ממעגל חיצוני שבאיור כאן מיוצג על ידי מייצב מתח של דיודת זנר וקבלי סינון . R1 נגד עבודה של דיודת הזנר.
 2. מתח VDD של ספק הכוח המתחבר לרכיב.
 3. מתח ייחוס ממייצב פנימי של 1.8v .
 4. מתח של 1.2v או מתח של 2.4v . קיים מייצב מתח פנימי של 1.2 וולט. מתח זה מגיע למגבר $1x/2x$ (מתואר על ידי משולש במרכז האיור) שהגברה שלו נשלטת על ידי ביט REFBGS ברגיסטר REF0CN . הביט קובע האם ההגברה היא X1 או 2X. במצב שבביט נשים 0 יש הגברה פי 2 ויגיע מתח ייחוס של 2.4V . ובמצב של 1 בביט יש הגברה של 1 ויגיע מתח ייחוס של 1.2V .
- בחלק התחתון במרכז האיור יש 2 מרבבים (מרבב 1 ומרבב 2) הנשלטים על ידי 2 ביטים ברגיסטר REF0CN . המרבב השמאלי – מרבב 1 – קובע האם ליציאה שלו תתחבר הכניסה העליונה שלו או הכניסה התחתונה שלו. הכניסה העליונה יכולה לקבל את מתח הייחוס מההדק החיצוני P1.5 או מחוצץ/מגבר. הכניסה התחתונה של מרבב 1 נמצא מתח ה VDD (שהוא 3.3 וולט). זאת ביט REFSL ברגיסטר ה REF0CN קובע מי מהשתיים תעבור ליציאה. המרבב הימני – מרבב 2 – קובע האם ליציאה שלו תחובר הכניסה העליונה המתחברת ליציאת מרבב 1 או מתח ייחוס פנימי של 1.8v . יציאת מרבב 2 נקבעת בעזרת ביט REGOVR . בצד ימין למעלה רואים 2 מלבנים :
- Bias Generator** – מחולל ממתח. זוהי מערכת היוצרת ממתח פנימי BIAS של 2.42v עבור רוב הרכיבים הפריפריאליים האנאלוגיים ברכיב כמו ה DAC , ה ADC , המתנדנים הפנימיים ברכיב וחיישן הטמפרטורה. הוא מופעל אוטומטית כאשר אחד הרכיבים הפריפריאליים האלו מאפשר ואין צורך להפעיל אותו במיוחד בעזרת הביט הזה. ניתן לראות שכאשר מפעילים את חיישן הטמפרטורה (בעזרת ביט TEMPEN=1) מחולל הממתח מקבל 1 דרך שער OR ומופעל. שער ה OR מתחבר גם לקו הנקרא IOSCEN . זהו ביט ברגיסטר OSCICN שמאפשר את פעולת המתנד הפנימי של 48 מגה הרץ (הרגיסטר לא בתוכנית הלימודים). כשנשים בביט '1' שער ה OR יעביר אותו למחולל והוא ייצור את הממתח .
- Temp Sensor** – שהוא קיצור של Temperature Sensor - חיישן טמפרטורה. ברכיב יש חיישן שבעזרתו ניתן למדוד את הטמפרטורה הפנימית של הרכיב.
- הרגיסטר שקובע את מתח הייחוס הוא REF0CN . reference CoNtrol – בקרת ייחוס . הרגיסטר נראה באיור שבהמשך :
- נסביר את תפקידי הביטים של הרגיסטר :
- ביט 7 – REFBGS** - REference Buffer Gain Select – בחירת הגבר חוצץ הייחוס . בעזרת ביט זה קובעים את מתח הייחוס של המייצב פנימי . אם שמים 0 אז ההגברה היא 2 ואם שמים 1 ההגבר הוא 1 .
- ביטים 5 ו 6 לא בשימוש .**
- ביט 4 – REGOVER** - REGulator Reference OVeride – ביטול (עקיפה) של מייצב הייחוס. ביט זה יכול לבטל את ביט 3 הנקרא REFSL . במצב 0 ביט 3 קובע מאיפה יגיע מתח הייחוס לממיר . במצב 1 מתח המייצב הפנימי יהיה מתח הייחוס.
- ביט 3 – REFSL** - REference SeLect – בחירת הייחוס . ביט זה קובע את מתח הייחוס . במצב 0 הדק VREF הוא מתח הייחוס ובמצב 1 מתח הייחוס הוא VDD .
- ביט 2 – TEMPE** - TEMperature sesor Enable – אפשרור חיישן הטמפרטורה. בתוך המיקרו יש חיישן טמפרטורה שבעזרתו ניתן לדעת את הטמפרטורה. כאשר נשים את הביט ב 0 חיישן הטמפרטורה איננו עובד. אם נשים 1 החיישן יעבוד. ניתן להכניס את מתח החיישן להמרה בעזרת המרבב AMUX0P .

SFR Definition 7.1. REF0CN: Reference Control

Bit	7	6	5	4	3	2	1	0
Name	REFBGS			REGOVR	REFSL	TEMPE	BIASE	REFBE
Type	R/W	R	R	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xD1; SFR Page = All Pages

Bit	Name	Function
7	REFBGS	Reference Buffer Gain Select. This bit selects between 1x and 2x gain for the on-chip voltage reference buffer. 0: 2x Gain 1: 1x Gain
6:5	Unused	Read = 00b; Write = don't care.
4	REGOVR	Regulator Reference Override. This bit "overrides" the REFSL bit, and allows the internal regulator to be used as a reference source. 0: The voltage reference source is selected by the REFSL bit. 1: The internal regulator is used as the voltage reference.
3	REFSL	Voltage Reference Select. This bit selects the ADCs voltage reference. 0: V_{REF} pin used as voltage reference. 1: V_{DD} used as voltage reference.
2	TEMPE	Temperature Sensor Enable Bit. 0: Internal Temperature Sensor off. 1: Internal Temperature Sensor on.
1	BIASE	Internal Analog Bias Generator Enable Bit. 0: Internal Bias Generator off. 1: Internal Bias Generator on.
0	REFBE	On-chip Reference Buffer Enable Bit. 0: On-chip Reference Buffer off. 1: On-chip Reference Buffer on. Internal voltage reference driven on the V_{REF} pin.

איור 18 : רגיסטר REF0CN.

ביט 1 – BIASE - internal analog generator BIAS Enable bit – ביט המאפשר להפעיל מחולל אנאלוגי פנימי של ממתח.

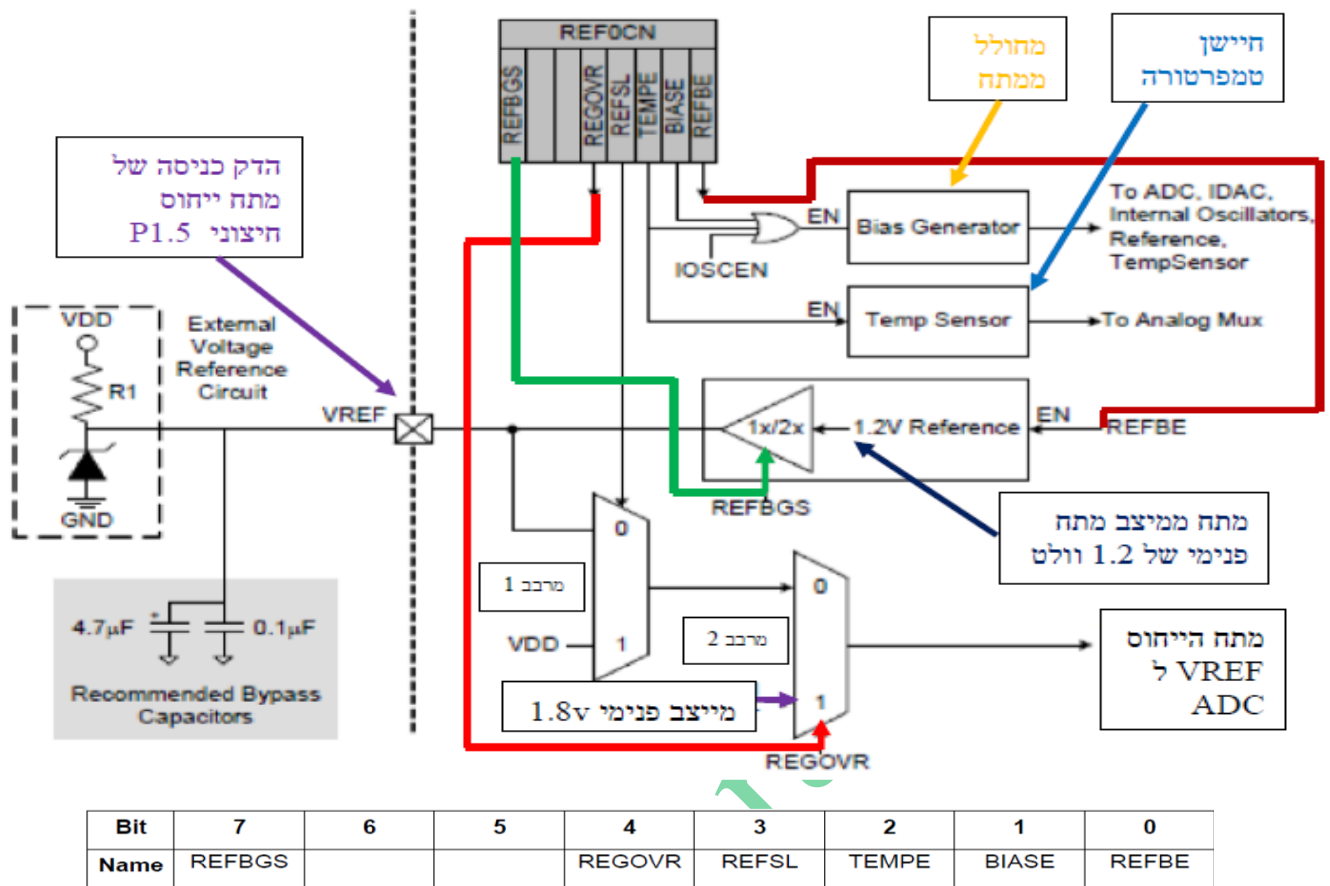
במצב 0 המחולל איננו עובד ובמצב 1 הוא עובד.

ביט 0 – REFBE - on chip REFERENCE Buffer Enable bit – ביט אפשרור חוצץ/מגבר ייחוס פנימי. במצב 0 החוצץ איננו

עובד. במצב 1 הוא עובד.

7.13.1 דוגמאות עבור רגיסטר REF0CN

בדוגמאות שבהמשך נסתמך על האיור הבא שבו איחדנו 2 איורים קודמים



איור 19 : העתק של 2 האיורים הקודמים .

דוגמה 1:

מהו הערך שנקבע ל REF0CN אם נרצה שמתח הייחוס יהיה מתח ייחוס חיצוני שמכניסים בהדק P1.5 של הרכיב.

פתרון:

REFBS - קובע את הגברת החוצץ/מגבר . לא משנה מה נשים בביט זה . יש לזכור לשים בביט 0 הנקרא **REFBE**

'0' כדי שהחוצץ/מגבר לא יעבוד ולא יתחבר אל הדק P1.5 . אחרת תהיה התנגשות בין מתח הייחוס החיצוני ומתח החוצץ.

REGOVR=0 כדי שהכניסה העליונה של מרבב 2 תעבור ליציאה.

REFSL=0 כדי שמרבב 1 יעביר את מתח ההדק P1.5 ליציאה שלו.

TEMPE=0 - לא מפעילים את חיישן הטמפרטורה כי הוא לא בשימוש בדוגמה שלנו וכך נחסוך בתצרוכת ההספק של הרכיב.

BIASE=0 - אין צורך להפעיל מחולל ייחוס .

ומכאן : **REF0CN = 0** .

דוגמה 2 :

מהו הערך שנקבע ל REF0CN אם נרצה שמתח הייחוס יהיה : א. VDD (מתח VDD=3.3v) . ב. מתח ייחוס פנימי של 1.2 וולט .

ג. מתח ייחוס פנימי של 1.8 וולט .

פתרון : בעזרת הניתוח עבור כל ביט של הדוגמה הקודמת נקבל :

א. REF0CN = 8 ב. REF0CN = 0x81 (אין להשתמש בהדק P1.5 !) ג. REF0CN = 0x10 .

7.14 תוכנה להפעלת הממיר

כדי להפעיל את הממיר יש לבצע 2 שלבים. שלב ראשון לאתחול את הממיר. זאת עושים בדרך כלל בתחילת התוכנית כאשר קובעים מצב של כל הדק (קלט או פלט) ומאתחלים רכיבים פריפריאליים שבמיקרו (טיימרים, פסיקות וכו'). בשלב השני לבצע קריאה.

7.14.1 אתחול הממיר

1. בהנחה שנרצה למדוד מתח יחסית לאדמה Single Ended נשלח ל 2 מרבבי ה AMUX, החיובי והשלילי, ערך מתאים. נניח שרוצים למדוד את הערך האנאלוגי של כניסה P1.0 יחסית לאדמה אז נרשום:

AMX0P = 0x15;

AMX0N = 0x1F;

2. יש לקבוע את ההדק כהדק אנאלוגי ולא דיגיטאלי (ברירת המחזל היא הדק דיגיטאלי). לשם כך נרשום ברגיסטר P1MDIN - Port 1 INput MoDe :

Bit	7	6	5	4	3	2	1	0
Name	P1MDIN[7:0]							
Type	R/W							
Reset	1*	1	1	1	1	1	1	1

אחרי RESET
בביט יש '1'
שאומר שההדק

דיגיטאלי. ביט שנשים '0' נקבע אותו כאנאלוגי. ומכאן :

P1MDIN = 0xFE ; // P1.0 = analog pin

כאשר הגדרנו את P1.0 כאנאלוגי אין צורך לקבוע את מצבו ברגיסטר P1MDOUT - Port 1 OUTput MoDe כי בעזרת רגיסטר זה קובעים האם ההדק הוא Open Drain =0 או Push Pull =1 ובביטים מ P1.1 עד P1.7 אנחנו לא משתמשים כאן.

3. הפעלת ה ADC בעזרת רגיסטר הבקרה ADC0CN.

נניח שרוצים שהתחלת ההמרה תהיה על ידי כתיבה לביט AD0BUSY.

Name	AD0EN	AD0TM	AD0INT	AD0BUSY	AD0WINT	AD0CM[2:0]		
	1	0	0	0	0	0	0	0

AD0EN=1 נפעיל את ביט האפשרות כדי שה ADC יעבוד.

AD0TM=0 : מצב עקיבה רגילה ולא מושהית.

AD0INT=0 : דגל שמראה שלא הסתיימה המרה. הביט עולה ל 1 בסיום המרה (ובנוסף יש בקשת פסיקה).

AD0BUSY=0 : מראה שהממיר איננו עסוק ולא מבצע המרה.

AD0WININT=0 : מראה שאין התאמה בין מתח הממיר וחלון המשווה.

AD0CM[2:0]=000 מראה שמקור "התחל המרה" – Start Conversion - יהיה כשנשים 1 בביט AD0BUSY.

ADC0CN = 0x80 ; ומכאן :

4. קביעת מתח הייחוס שיגיע לממיר בעזרת הרגיסטר **REF0CN**. אם נרצה מתח ייחוס של 1.8 וולט נרשום **REF0CN=0x10**; (כמו בדוגמה בהסבר על הרגיסטר).
5. רגיסטר **ADC0CF** – לא נשנה את ברירת המחדל של הרגיסטר.

ADC0CF: ADC0 Configuration

Bit	7	6	5	4	3	2	1	0
Name	AD0SC[4:0]					AD0LJST	Reserved	
Type	R/W					R/W	R/W	
Reset	1	1	1	1	1	0	0	0

הביטים 3 עד 7 קובעים את תדר הפולסים שמגיעים ל SAR. חלוקת התדר של ה SYSCLK מתבצעת בביטים האלה. ברירת המחדל היא 1111 שאומרת שהמספר הוא 31. נשתמש בנוסחה שפיתחנו בהסבר על הרגיסטר:

$$CLK_{SAR} = SYSCLK / (AD0SC + 1)$$

ומכאן שהתדר שיגיע ל SAR עבור $STCLK = 12MHz$ הוא:

$$CLK_{SAR} = 12 \cdot 10^6 / (31+1) = 375,000$$

זמן מחזור של הפולסים $1/375000 = 2.666666666666667e-6$ (בערך 2.66 מיקרו שנייה).

צריך 11 פולסים לביצוע ההמרה (בעקיבה רגילה ולא מושהית) ולכן זמן ההמרה הוא: $t = 11 \cdot 2.66\mu Sec = 29.26 \mu Sec$.

7.14.2 הפעלת המרה וקריאת הערך הדיגיטאלי.

1. כותבים '1' לביט AD0BUSY ברגיסטר ה **ADC0CN** שאומר לו "התחל המרה". הממיר מתחיל לבצע המרה.

בשפת C51: **AD0BUSY = 1;**

2. אם רוצים לעבוד עם פסיקה של הממיר (פסיקה מספר 10) כותבים פונקציית פסיקה הקוראת את הנתון ברגיסטר ADC0. אם רוצים לעבוד עם שאילתה עושים POLLING על הביט AD0INT (ביט 5 ברגיסטר **ADC0CN**). בשפת C51 נרשום **while(AD0INT==0);** (אפשר כמובן לרשום גם **while(!AD0INT);**), כלומר נמצאים בלולאה כל עוד הביט ב 0 כי עדיין לא הסתיימה ההמרה. אחרי 11 פולסים (או 14 פולסים במצב עקיבה מושהית) מסתיימת ההמרה וביט AD0INT עולה ל 1 ויש בקשת פסיקה של ה ADC. אם עובדים עם פסיקה עוברים לפונקציית הפסיקה. אם עובדים עם שאילתה מבצעים קריאה של הנתון מיד בפקודה הבאה.

נגדיר משתנה בשם value מטיפוס שלם:

כתיבת '1' לביט שאומרת "התחל המרה" **AD0BUSY = 1;** //

while(AD0INT==0); ממתינים שההדק יעלה ל 1 בסיום ההמרה **while(!AD0INT);** //

ניקוי הביט להכנה למחזור ההמרה הבא **AD0INT = 0;** //

הכנסת קריאת הממיר למשתנה value **value=ADC0;** //

הערה: הפקודה האחרונה **value=ADC0;** עובדת כי בקובץ הכותר REG51F380.H של חברת SiLABS או בקובץ הכותר

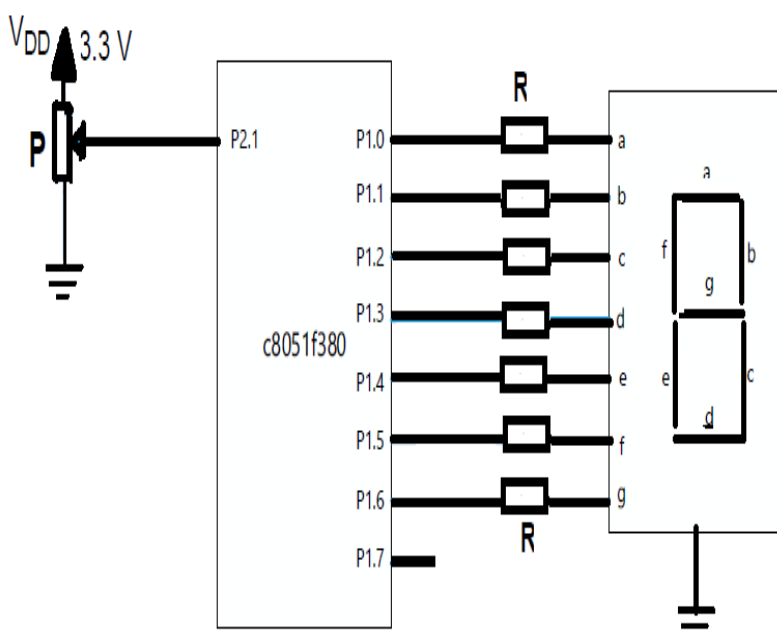
C8051F380_defs.h של חברה אחרת שנכלול בתחילת התוכנית שלנו נרשמה השורה **SFR16 (ADC0, 0xBD);**

שאומרת ש ADC0L נמצא בכתובת 0xBD באזור ה SFR ו ADC0H נמצא בכתובת הבאה 0xBE וניתן להתייחס אליהם כרגיסטר אחד של 16 ביטים. אם לא הייתה נתונה השורה הזו היינו צריכים לרשום: $value = 256 * ADC0H + ADC0L$;

7.15 דוגמאות להפעלת הממיר

דוגמה 1 :

נתון האיור הבא שבו רואים מצד שמאל מיקרו בקר C8051F380 המחובר לפוטנציומטר ותצוגת 7 מקטעים (דרך נגדי R). המיקרו מודד את המתח של זחלן הפוטנציומטר P המתקבל בהדק P2.1 בעזרת ה ADC0 שלו ומציג את הקריאה של 3 רמות המתח האפשריות בתצוגת 7 המקטעים לפי הטבלה שבימין האיור. הנח שההמרה מתחילה על ידי כתיבה ל AD0BUSY. יש לרשום תכנית שתציג בתצוגת 7 המקטעים את מתח הזחלן לפי 3 רמות של מתח הכניסה.



$V_{P2.1}$	הצגה - 7Segment
$V_{P2.1} < 1V$	
$1V \leq V_{P2.1} \leq 2.5V$	
$V_{P2.1} > 2.5V$	

איור 20 : קריאת מתח של פוטנציומטר.

פתרון דוגמה 1 :

- המתח המגיע מהזחלן הוא 3.3 וולט כאשר הזחלן למעלה ו 0v כאשר הזחלן למטה. מתח 3.3v מאלץ אותנו לבחור מתח ייחוס לממיר של VDD שהוא 3.3V כי שאר מתחי הייחוס שיכולים להגיע לממיר קטנים יותר. הסיבה לכך שאם נבחר את מתח הייחוס השני בגודלו 2.4 וולט - אז ממתח זה עד 3.3 וולט הקריאה שנקבל היא 0X3FF.
- הרזולוציה במקרה שלנו היא :

$$RESolution = 3.3 / 2^{10} = 0.003222v = 3.222mv = RES$$

- הקריאה שנקבל מהממיר עבור מתח כניסה של 1v היא :

$$1 / RES = 310.36 \approx 310$$

הקריאה שנקבל עבור מתח כניסה של 2.5v היא :

$$2.5 / \text{RES} = 775.915 \approx 776$$

בתצוגת 7 המקטעים יש להציג 3 תווים לפי הטבלה הבאה:

ערך HEX	a	b	c	d	e	f	g	X	התו להצגה	ערך מתח
38	0	0	0	1	1	1	0	0	L	מתחת ל 1v
40	0	0	0	0	0	0	1	0	-	בין 1 ל 2.5v
76	0	1	1	0	1	1	1	0	H	מעל 2.5v

`unsigned char displayTable[] = {0X38, 0X40 , 0X76 };`

את הטבלה הזו נכניס למערך :

4. היות ורוצים למדוד את מתח זחלן הפוטנציומטר יחסית לאדמה - Single Ended - נשלח ל 2 מרבבי ה AMUX החיובי והשלילי, את הערכים :

`AMX0P = 0x01;`

`AMX0N = 0X1F;` // חיבור ההדק השלילי של הממיר לאדמה

5. יש לקבוע את ההדק כהדק אנאלוגי ולא דיגיטאלי (ברירת המחדל היא הדק דיגיטאלי). לשם כך נרשום ברגיסטר P2MDIN - Port 2 INput MoDe :

Bit	7	6	5	4	3	2	1	0
Name	P2MDIN[7:0]							
Type	R/W							
Reset	1	1	1	1	1	1	1	1
	1	1	1	1	1	1	0	1

אחרי RESET בביטים יש '1' שאומר שההדקים דיגיטאליים. ביט שנשים '0' נקבע אותו כאנאלוגי. ומכאן :

`P2MDIN = 0xFD ;` // P2.1 הדק אנאלוגי

כאשר הגדרנו את P2.1 כאנאלוגי אין צורך לקבוע את מצבו ברגיסטר P2MDOUT - Port 2 OUTput MoDe כי בעזרת רגיסטר זה קובעים האם ההדק הוא Open Drain =0 או Push Pull =1 ובביטים מ P2.0 ו P2.2 עד P2.7 אנחנו לא משתמשים כאן.

6. הפעלת ה ADC בעזרת רגיסטר הבקרה ADC0CN .

נניח שרוצים שהתחלת ההמרה תהיה על ידי כתיבה לביט AD0BUSY .

Name	AD0EN	AD0TM	AD0INT	AD0BUSY	AD0WINT	AD0CM[2:0]		
	1	0	0	0	0	0	0	0

AD0EN=1 נפעיל את ביט האפשרות כדי שה ADC יעבוד.

AD0TM=0 : מצב עקיבה רגילה ולא מושהית.

AD0INT=0 : דגל שמראה שלא הסתיימה המרה. הביט עולה ל 1 בסיום המרה (ובנוסף יש בקשת פסיקה).

AD0BUSY=0 : מראה שהממיר איננו עסוק ולא מבצע המרה.

AD0WININT=0 : מראה שאין התאמה בין מתח הממיר וחלון המשווה.

AD0CM[2:0]=000 מראה שמקור "התחל המרה" – Start Conversion – יהיה כשנשים 1 בביט AD0BUSY.

ADC0CN = 0x80 ;

ומכאן :

7. קביעת מתח הייחוס שיגיע לממיר בעזרת הרגיסטר **REF0CN**. אם נרצה מתח ייחוס של VDD (3.3 וולט) נרשום

REF0CN=0x08 ;

8. ברגיסטר **ADC0CF** – לא נשנה את ברירת המחדל של הרגיסטר.

9. נגדיר את P1 שמפעיל את תצוגת 7 המקטעים כפלט עם עבודה במצב PUSH-PULL. לשם כך יש לכתוב לרגיסטר

P1MDOUT את הערך הבא:

Bit	7	6	5	4	3	2	1	0
Name	P1MDOUT[7:0]							
Type	R/W							
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xA5; SFR Page = All Pages

Bit	Name	Function
7:0	P1MDOUT[7:0]	Output Configuration Bits for P1.7–P1.0 (respectively). These bits are ignored if the corresponding bit in register P1MDIN is logic 0. 0: Corresponding P1.n Output is open-drain. 1: Corresponding P1.n Output is push-pull.

נשים בכל הביטים 1 כדי לקבוע שכל ההדקים של פורט 1 הם יציאות עם PUSH-PULL כדי שיוכלו לתת זרם לסגמנטים של התצוגה.

P1MDOUT = 0xFF ;

התוכנית תיראה כך :

```
#include "compiler_defs.h"
#include "C8051F380_defs.h"
// ----- הצהרות על פונקציות -----
void init ( ); // הצהרה על פונקציה המאתחלת את הממיר וכמו כן את פורט 1 לפלט
int readADC0(void); // הצהרה על הפונקציה המחזירה ערך המתאים למתח הכניסה
//----- הגדרת משתנים גלובאליים -----
int val; // למשתנה הזה ייכנס הערך של המתח הנבדק
int readTable[ ] = { 310 , 776 }; // טבלת הערכים הדיגיטאליים המתאימה ל 1 וולט ו 2.5 וולט
unsigned char displayTable[ ] = { 0X38, 0X40 , 0X76 }; // הערכים שנשלח לתצוגת 7 המקטעים
```

//----- התוכנית הראשית -----

void main ()

{

initADC0(); // פונקציה לאתחול הממיר ולאתחול פורט 1 כפלט

while (1) // לולאה אין סופית הממירה את מתח הזחלון ומציגה בהתאמה 3 רמות מתח בתצוגת 7 המקטעים

{

val = readADC0(); // val לערך דיגיטאלי ומחזירה את הערך ל val

if (val < 310) // האם הערך הדיגיטאלי קטן ממתח אנלוגי של 1 וולט ?

P1=displayTable[0]; // L להדליק בתצוגה

else if (val >776) // האם הערך הדיגיטאלי גדול ממתח אנלוגי של 2.5 וולט ?

P1= displayTable[2]; // H להדליק בתצוגה

else // הערך הדיגיטאלי גדול או שווה ל 1 וולט וקטן או שווה ל 2.5 וולט

P1=displayTable[1]; // להדליק בתצוגה קו באמצע -

}

}

void init () // פונקציה המאתחלת את הממיר וכמו כן את פורט 1 לפלט

{

AMX0P = 0x01; // הכנסת הדק P2.0 לממיר לכניסת ה + שלו

AMX0N = 0x1F; // חיבור ההדק השלישי של הממיר לאדמה

P2MDIN = 0xFD ; // הדק אנלוגי P2.1

ADC0CN = 0x80 ; // "התחל המרה" יהיה כשנשים 1 בביט AD0BUSY

REF0CN=0x08; // מתח ייחוס של (3.3 וולט) VDD

P1MDOUT = 0xFF ; // הדקי פורט 1 הם יציאות עם PUSH-PULL

XBR1=0x40; // ביט לאפשר הקרוסבר xbare =1

}

int readADC0() // הפונקציה המחזירה ערך המתאים למתח הכניסה

{

AD0BUSY = 1; // כתיבת '1' לביט שאומרת "התחל המרה"

while (!AD0INT); // ממתינים שההדק יעלה ל 1 בסיום ההמרה while(AD0INT==0);

AD0INT = 0; // ניקוי הביט להכנה למחזור ההמרה הבא

return (ADC0);

}

דוגמה 2

לפעמים רוצים להמיר מספר מתחים שונים ולא מתח אחד בלבד כמו בדוגמה הקודמת. אילו שינויים יש להוסיף בתוכנית? רשום פונקציה שתקבל את מספר ההדק הרצוי ותחזיר את הערך הדיגיטאלי המתאים למתח הנמדד בהדק זה. פתרון דוגמה 2:

נניח שרוצים למדוד מתח אנאלוגי הנכנס בהדק P3.2.

דבר ראשון יש לקבוע הדק זה כהדק אנאלוגי בעזרת P3MDIN.

```
P3MDIN = 0XFB; // 11111011
```

בשלב השני נסתכל בדפי הנתונים של AMX0P (בדפי הנתונים של הרכיב או בהסבר על הרגיסטרים של הממיר בספר כאן) ונראה שהדק זה מתאים ל-011011 (שני הביטים הגבוהים הם לא בשימוש. כלומר הכתובת של הדק זה בכניסת המרבב האנאלוגי היא: 0X1B. השורה הקוראת תראה:

```
val = readADC0(0x1B);
```

הפונקציה עצמה דומה לזו שבתרגיל הקודם אבל בתחילת הפונקציה מכניסים לממיר את הכתובת המתאימה של ההדק הרצוי (P3.2). כמו כן יש להצהיר על פונקציית ההשהיה delay_us(int us); שמקבלת ערך שלם ומשהה את התוכנית לפי הערך שקיבלה במיקרו שניות. הפונקציה של הקריאה מהממיר תראה עכשיו כך:

```
int readADC0( int address) // הפונקציה המחזירה ערך המתאים למתח הכניסה
{
    AMXP=address; // אומרים למרבב האנאלוגי החיובי להכניס את המתח בהדק P3.2 להדק החיובי של הממיר
    delay_us(5); // השהייה קצרה כדי שהמרבב האנאלוגי יחליף מצב והקבל יעקוב אחרי המתח בהדק (יספיק להיטען).
    AD0BUSY = 1; // כתיבת '1' לביט שאומרת "התחל המרה"
    while (!AD0INT); // ממתינים שההדק יעלה ל-1 בסיום המרה
    AD0INT = 0; // ניקוי הביט להכנה למחזור ההמרה הבא
    return (ADC0);
}
```

7.16 ביבליוגרפיה

1. דפי נתונים מהאתר של SILABS [C8051F38x Full Speed USB Flash MCU Family Data Sheet \(silabs.com\)](http://silabs.com)
2. [SAR: Successive Approximation Register \(silabs.com\)](http://silabs.com)
3. Silicon Labs, אבי חיון ושי מזל, חלק א, הוצאת ידעטק.
4. אתר מוקד מקצוע [שאלות דוגמה בבחינת מיקרו-בקר ממשפחת סיליקון לאב \(education.gov.il\).pdf](http://www.education.gov.il)