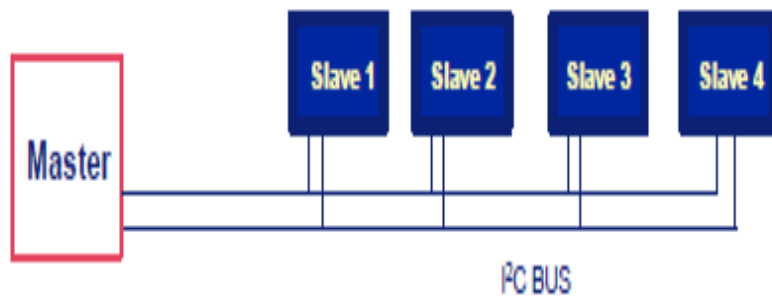


## תקשורת I2C

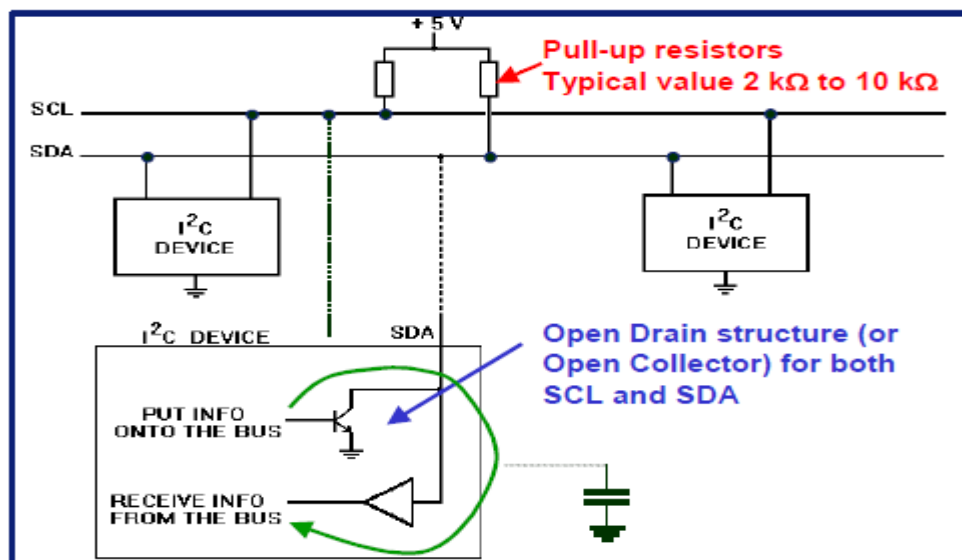
תקשורת I<sup>2</sup>C נקראת גם IIC או I2C היא תקשורת טורית בין מעבד- מסטר – MASTER ובין רכיב עבד – SLAVE. היא פותחה בתחילת שנות ה 80 על ידי חברת פיליפס שנקראת היום NXP semiconductors. על פס תקשורת I<sup>2</sup>C יכולים להתחבר מספר רכיבים שונים (זיכרונות, ממירים, שעוני זמן אמת, מרחיבי פס וכו'). הרכיב המנהל את תהליך התקשורת (המעבד) נקרא MASTER והרכיבים המתחברים אליו נקראים SLAVES. בתקשורת זו ישנם שני קווים. קו הנתונים הטורי – SDA (Serial Data) - שהוא דו כיווני וקו השעון הטורי - SCLK (Serial CLock) - שהוא חד כיווני ומופעל על ידי ה MASTER. בנוסף, ה MASTER שולט על הגישה לפס ויוצר את מצבי ה START (התחלה) וה STOP (סיום).

איור 1 א' מתאר מספר רכיבים המתחברים על קו התקשורת I<sup>2</sup>C



איור 1 - חיבור של מספר רכיבי SLAVE אל MASTER

באיור 1 ניתן לראות 4 רכיבי SLAVE המתחברים אל MASTER. באיור 2 יש פרוט של נגדי ה PullUp וכיצד נראית דרגת היציאה והכניסה של רכיב המתחבר בתקשורת I<sup>2</sup>C.



איור 2 - קו תקשורת I<sup>2</sup>C מפורט

ניתן לראות שעל 2 הקווים SDA (קו הנתון) ו SCL (קו השעון) יכולים להתחבר מספר רכיבים. לכל רכיב יש כתובת ייחודית משלו. לרכיב DS1307 הכתובת היא 68H (וכאשר מזיזים ביטים פעם אחת שמאלה – יוסבר מדוע בהמשך) הכתובת היא 1101000X (D0H לכתיבה או D1H לקריאה). לרכיב קול של חברת WINBOND הנקרא ISD5116 הכתובת היא 40H או 80H לאחר הזזה של ביט שמאלה וכך הלאה.

באיור רואים 2 רכיבים המתחברים על הקווים. בחלק התחתון של האיור רואים מבנה פנימי של רכיב ורואים שהרכיב מתחבר בעזרת חוצץ (מתואר על ידי המשולש) המקבל נתון מהקו. מעל החוצץ יש טרנזיסטור בחיבור קולט פתוח (Open Collector) או טרנזיסטור תופעת שדה - FET - בחיבור מפק פתוח (Open Drain), שיכול לכתוב לקו נתון. לטרנזיסטור יש לחבר נגד חיצוני שערכו נע בין 2 קילו אוהם ל 10 קילו אוהם. הערכים נבחרים כך שמצד אחד הנגדים לא יהיו קטנים מידי כדי שלא יזרום זרם גדול דרך הקווים ודרך הרכיב (במצב שהרכיב מוציא 0) ומצד שני שהנגד לא יהיה גדול מידי כי הוא קובע את זמן הטעינה והפריקה במעברים בין 0 ל 1 ולהפך ונגד גדול מידי יגביל את קצב התקשורת.

## 1. כללים והגדרות בתקשורת I<sup>2</sup>C

- העברה נתון יכולה להתחיל רק כאשר הקו לא עסוק - NOT BUSY.
- בזמן העברת נתון, קו הנתון חייב להישאר יציב כאשר קו השעון במצב גבוה. שינוי בקו הנתון כאשר קו השעון הוא גבוה יתפרש כאות בקרה.

מגדירים את מצבי הפס הבאים:

### Bus Not Busy - פס לא עסוק

גם קו הנתון וגם קו השעון בגבוה.

### START DATA TRANSFER - התחל העברת נתון

שינוי במצב קו הנתון מגבוה לנמוך כאשר השעון נמצא בגבוה מוגדר כמצב START.

### STOP DATA TRANSFER - עצור העברת נתון

שינוי במצב קו הנתון מנמוך לגבוה כאשר השעון במצב גבוה מוגדר כמצב STOP.

### DATA VALID - תקפות נתון

מצב קו הנתון מייצג תקפות נתון כאשר לאחר מצב START, קו הנתון יציב למשך זמן הגבוה של אות השעון. הנתון בקו חייב להשתנות רק בזמן מצב נמוך של אות השעון. יש פולס שעון אחד עבור כל ביט של נתון.

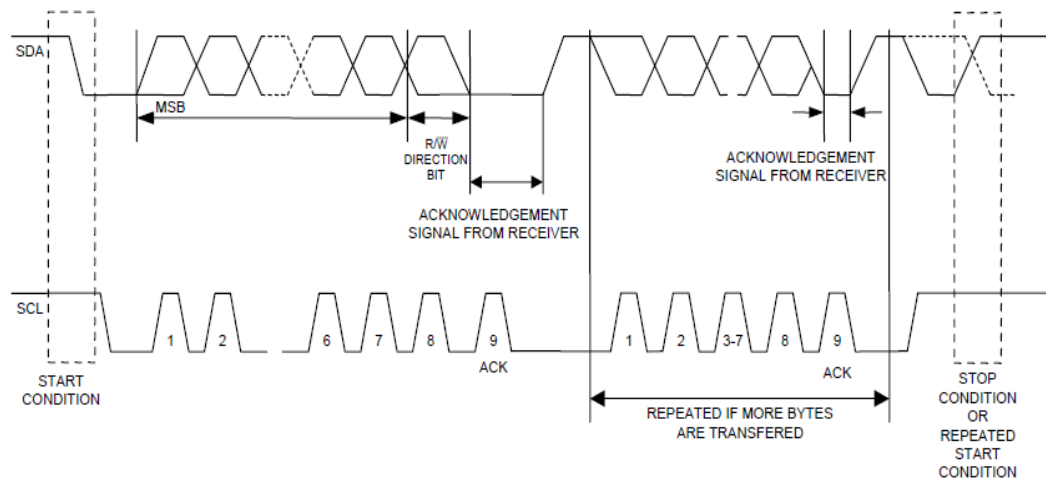
כל העברת נתונים מתחילה עם מצב START ומסתיימת עם מצב STOP. כמות הבתים המועברת בין START ל STOP לא מוגבלת ונקבעת על ידי רכיב ה MASTER. האינפורמציה מועברת ביית אחרי ביית וכל מקלט מאשר קבלת הבית עם ביט תשיעי של ACKNOWLEDGE.

בהגדרות של I<sup>2</sup>C יש תקן של קצב ב 100KHz ויש תקן ל 400KHz.

## ACKNOWLEDGE – אישור

כל רכיב קולט חייב בסיום קליטת ביית, שהועבר אליו, ליצור ביט ACKNOWLEDGE. רכיב ה-MASTER יוצר פולס שעון נוסף הקשור לביט זה.

רכיב היוצר ACKNOWLEDGE חייב להוריד את קו הנתון הטורי – SDA – ל 0 בזמן פולס השעון, כלומר שקו הנתון יהיה יציב בנמוך בזמן שקו השעון בגבוה. רכיב ה-MASTER מסמן ל-SLAVE על סיום התקשורת על ידי **אי יצירת** ביט ה-ACKNOWLEDGE כאשר הוא קלט את הביית האחרון מה-SLAVE. במקרה כזה על ה-SLAVE להשאיר את קו הנתון בגבוה כדי לאפשר ל-MASTER ליצור מצב STOP. באיור 3 ניתן לראות העברה של נתון טורי.



**איור 3** - העברת נתון בקו תקשורת טורית I<sup>2</sup>C

את הקו SCL (הקו התחתון בשרטוט) של פולסי השעון יוצר תמיד ה-MASTER. כדאי לשים לב שמצב START קורה כאשר קו SCL בגבוה ואז ה-MASTER הוריד את קו הנתון ל 0. לאחר מכן ה-MASTER יוצר 8 פולסי שעון ואז הוא שולח בקו הנתון - SDA - 8 ביטים. 7 ביטים הם כתובת הרכיב והביט ה-8 אומר האם הוא רוצה לכתוב אל הרכיב או לקרוא ממנו (0 - כתיבה, 1 - קריאה). לאחר מכן ה-MASTER יוצר פולס 9 נוסף שבו ה-SLAVE צריך להחזיר ACKNOWLEDGE. לאחר מכן אין צורך ב-START נוסף והבתים נשלחים אחד אחרי השני כאשר הצד הקולט נותן ACKNOWLEDGE בביט מספר 9. מצב STOP (או START חוזר) מתואר בצד ימין של איור 6. הוא נוצר כאשר קו השעון ב 1 ואז בקו הנתון יש מעבר מ 0 ל 1. מצב START חוזר משורטט בקו מקווקו ובו רואים שבזמן שקו השעון ב 1 יורד קו הנתון ל 0.

## 2. העברת נתון בתקשורת I<sup>2</sup>C

שתי אפשרויות העברת נתונים קיימות בקו תקשורת I<sup>2</sup>C :

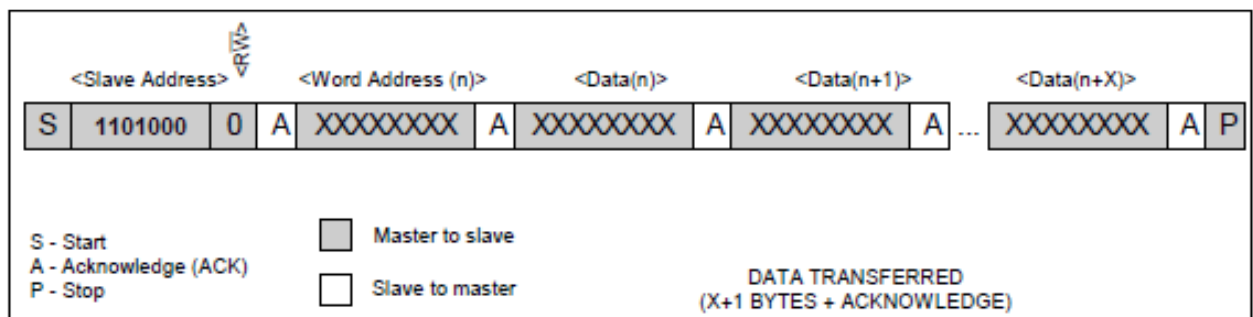
**א.** ה-MASTER משדר וה-SLAVE קולט - אופן כתיבה - Write Mode

במקרה זה הביית הראשון המשודר על ידי ה MASTER הוא ה **כתובת** של ה SLAVE המכילה 7 ביטים מביט D1 עד D7 ועוד ביט LSB ( ביט D0 ) המציין האם מבצעים כתיבה ( בביט יש 0 ) או קריאה ( בביט יש 1 ). לדוגמה במקרה של רכיב DS1307 או DS3231 שהם רכיבי שעון זמן אמת – RTC - הכתובת היא 0110100X - 0x68 במקרה של **כתיבה** לרכיב או 0x69 אם **קוראים** מהרכיב . לאחר מכן יבואו מספר בתים של נתונים. ה SLAVE מחזיר ACKNOWLEDGE בסיום כל ביית נתונים שקלט. העברת הנתון מתחילה מביט ה **MSB** ראשון ועד ביט ה **LSB** !!

**ב. ביית משודר מה SLAVE אל ה MASTER - אופן קריאה - Read Mode**

במקרה זה הביית הראשון שנשלח הוא על ידי ה MASTER השולח את כתובת ה SLAVE שמחזיר מצידו את ביט ה ACKNOWLEDGE . מכאן ה SLAVE שולח מספר בתי נתונים. ה MASTER מחזיר ביט ACKNOWLEDGE אחרי כל קליטת ביית נתון חוץ מהביית האחרון שהוא איננו מחזיר ACNOWLEDGE או אפשר להגיד שהוא מחזיר Not ACKNOWLEDGE .

**3. אופן כתיבה – ה MASTER משדר אל אחד מה SLAVES**



**איור 4 - אופן כתיבת נתון מה MASTER כשה SLAVE הוא המקלט .**

באיור 4 מתואר מצב שבו ה MASTER כותב אל ה SLAVE המשמש כמקלט. החלק הכהה שבאיור הוא מה ששולח ה MASTER. החלק הבהיר הוא המסומן כ A הוא ה Acknowledge – אישור - ששולח ה SLAVE . ה MASTER יוצר מצב START (מסומן ב S ). לאחר מכן הוא שולח 7 ביטים של כתובת הרכיב - DOH במקרה של הרכיב DS1307 - והביט ה 8 הוא 0 המציין שהוא הכותב וה SLAVE הוא המקלט. על ה SLAVE לענות ב ACKNOWLEDGE (מסומן ב A ). לאחר מכן ה MASTER שולח ביית נוסף הטוען את מצביע (אוגר) הכתובות בתוך הרכיב. הנתון הבא נכתב לכתובת זו ומצביע הכתובות גדל אוטומטית ב 1 . כל נתון נכתב בכתובת שבמצביע הכתובות ומצביע הכתובות מתקדם ב 1 . אחרי כל ביית שנקלט על ידי ה SLAVE הוא שולח אישור שקלט – ACKNOWLEDGE . ה MASTER מסיים את התקשורת בעזרת מצב STOP (מופיע בצד ימין עם האות P ).

## 4. אופן קריאה - ה SLAVE משדר אל ה MASTER

גם מצב זה מתחיל תמיד במצב שבו ה MASTER משדר אל ה SLAVE אבל כאן הוא אומר שהוא רוצה לקרוא ממנו ( הוא שם בביט ה LSB של הכתובת 1 ). הביית הראשון שה MASTER משדר נקלט על ידי ה SLAVE כמו שתואר בפסקה הקודמת, כלומר ה MASTER ייצור מצב START , יישלח את 7 הביטים של הכתובת 10110001 אבל בביט השמיני – LSB - יהיה 1 שבו הוא אומר שהוא רוצה לקרוא. מכאן ה SLAVE משדר את הנתונים וה MASTER עונה עם ביט ACKNOWLEDGE . בביית האחרון , כשה MASTER רוצה בסיום התקשורת , הוא איננו מגיב בביט ה 9 ולא שולח ACKNOWLEDGE (מסומן באיור ב  $\bar{A}$  – not Acknowledge ) וגם שולח ביט עשירי של STOP .

כאשר רוצים לקרוא מרכיב SLAVE שיש לו זיכרון שבהם יש מספר נתונים אז ה MASTER כותב 3 בתים אל הרכיב. **בביית הראשון** ה MASTER אומר לרכיב ה SLAVE שהוא פונה אליו לכתיבה ( בביט ה LSB שמים 0 ) .

ה SLAVE נותן A – אישור .

**בביית השני** הוא מציין את הכתובת בזיכרון ה SLAVE ממנה הוא רוצה להתחיל לקרוא . זוהי הכתובת שבזיכרון ה SLAVE והיא של 8 ביטים. פעולה זו באה לטעון ב SLAVE רגיסטר מונה כתובות (שנקרא גם מצביע כתובות) .

ה SLAVE נותן A – אישור .

מיד לאחר מכן ה MASTER שולח START נוסף.

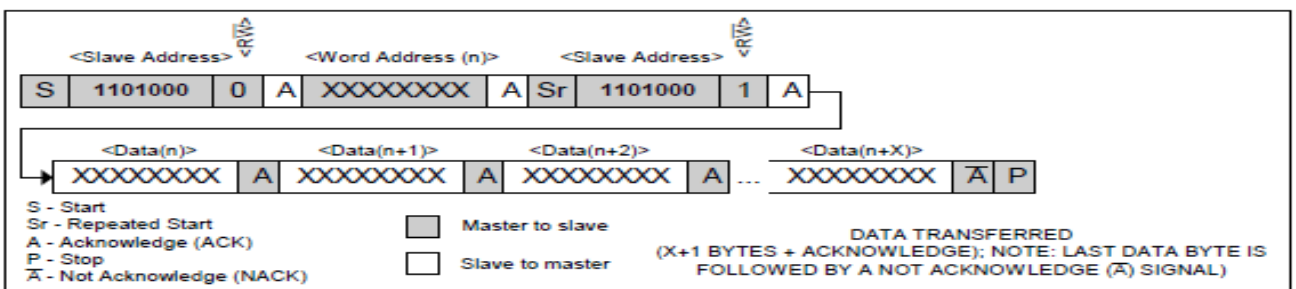
**בביית השלישי** ה MASTER שולח את כתובת ה SLAVE אבל בביט ה LSB יש 1 לציין קריאה .

ה SLAVE נותן אישור – A .

ה MASTER הופך את קו הנתון הטורי SDA לקלט.

מכאן ה SLAVE מתחיל לשלוח את הנתון מהכתובת שקיבל בבית השני וכל אישור - A מה MASTER יגרום למונה הכתובות שלו להתקדם ולשלוח את הנתון מהכתובת הבאה עד שלא יקבל אישור מה MASTER ולאחריו S המציין Stop – עצירה ואז התקשורת מסתיימת. מונה הכתובות ב SLAVE נשאר על הכתובת של הנתון הבא.

איור 5 מתאר פעולת קריאה מהכתובת הרצויה. גם כאן הצבע הכהה הוא של ה MASTER והבהיר של ה SLAVE .



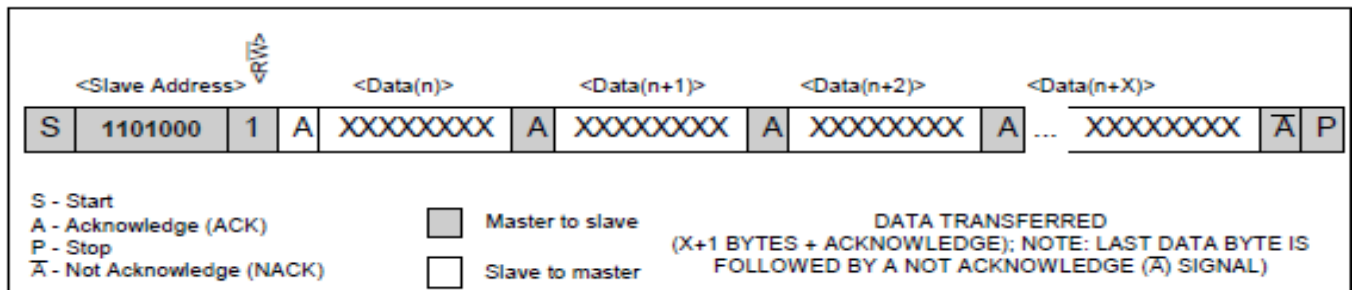
איור 5

– פעולה משולבת של כתיבה ל SLAVE כדי לציין כתובת רצויה וקריאה מה SLAVE – מהאיור רואים שה MASTER שולח 3 בתים אל ה SLAVE . בית ראשון הוא כתובת רכיב ה SLAVE בפס I2C , הביית השני הוא הכתובת בזיכרון של ה SLAVE ממנה ה MASTER רוצה לקבל נתון . מיד לאחר מכן ה MASTER יצר מצב START חוזר (מסומן ב Sr ), שלח בית נוסף שבו יש את כתובת הרכיב 01101001 ( הביט השמיני הוא של קריאה ) ומרגע זה ה SLAVE

משדר וה MASTER מגיב ב ACKNOWLEDGE . בסיום התקשורת ה MASTER לא נותן ACKNOWLEDGE (מסומן ב  $\hat{A}$ ) ולאחר מכן נותן מצב STOP .

הנתונים המשודרים מה SLAVE מתחילים מהכתובת ששלח אליו ה MASTER . אם ה MASTER לא שולח כתובת אז הנתון שמתקבל יהיה מהכתובת האחרונה שבה נמצא מצביע הכתובות של ה SLAVE . כל נתון שה SLAVE שולח הוא מקדם את מצביע הכתובות שלו לכתובת הבאה ושולח את הנתון מכתובת זו וכל עוד הוא מקבל אישור Acknowledge מהמסטר הוא משדר את הנתון הבא.

איור 6 מתאר מצב תקשורת זה. גם כאן הצבע הכהה הוא של ה MASTER והבהיר של ה SLAVE .



**איור 6** - אופן קריאה ללא ציון כתובת ב SLAVE .

מהאיור רואים שה MASTER שולח מצב START ומיד אחריו ביית שמציין את כתובת ה SLAVE בפס I2C עם 1 בביט ה LSB המציין קריאה. מרגע זה ה MASTER הופך את קו הנתון SDA לקלט וקולט את הביית ששולח ה SLAVE . כל נתון הנקלט על ידי ה MASTER נענה ב A – אישור של ה MASTER ואז ה SLAVE שולח ביית חדש. כאשר ה MASTER רוצה סיום תקשורת הוא לא נותן אישור – מצב  $\hat{A}$  ואז שולח מצב Stop והתקשורת מסתיימת.