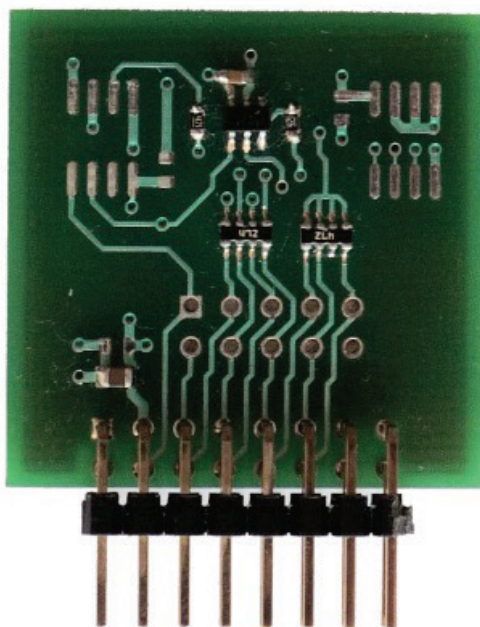


LOGSYS HŐMÉRŐ ÉS EEPROM MODUL

FELHASZNÁLÓI ÚTMUTATÓ



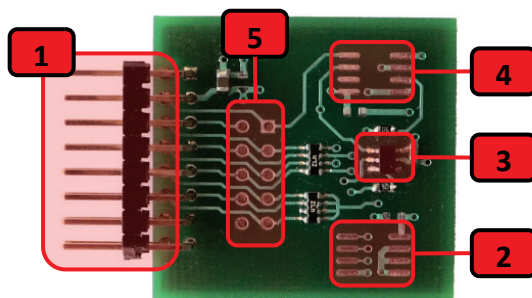
Tartalomjegyzék

1	Bevezetés.....	1
2	Az I²C busz általános ismertetése	2
3	Az SPI busz általános ismertetése	4
4	LM75 – I²C buszos hőmérő	5
5	TMP121 – SPI buszos hőmérő	9
6	M95080 – SPI buszos EEPROM	10
6.1	<i>Az EEPROM SPI interfésze</i>	<i>10</i>
6.2	<i>Az EEPROM státusz regisztere</i>	<i>11</i>
6.3	<i>Az EEPROM parancskészlete.....</i>	<i>12</i>
7	A modul kapcsolási rajza	15
	Változások a dokumentumban.....	16

1 Bevezetés

A bővítő modul kifejlesztésének fő célja az volt, hogy a hallgatókat megismertessük az alapvető, beágyazott rendszerekben napi szinten használt I²C és SPI soros busz szabványokkal. A modul felépítését az 1-1. ábra szemlélteti, a bővítőcsatlakozó lábkiosztása az 1-1. táblázatban látható. A modulon három integrált áramkör található:

- LM75 I²C buszos hőmérő chip
- TMP121 SPI buszos hőmérő chip
- M95080 SPI buszos EEPROM chip



1-1. ábra: A LOGSYS hőmérő és EEPROM modul.

A modul felépítése:

1. LOGSYS bővítőcsatlakozó
2. LM75 I²C buszos hőmérő chip (nem mindig van beültetve)
3. TMP121 SPI buszos hőmérő chip
4. M95080 SPI buszos EEPROM chip (nem mindig van beültetve)
5. CON2 csatlakozó (tesztpontok, lásd az 1-2. táblázatot)

1-1. táblázat: A bővítőcsatlakozó lábkiosztása.

Láb	Típus ¹	Funkció
1	PWR	GND (föld)
2	PWR	Nem használt (5V)
3	PWR	3,3V tápfeszültség
4	I	SPI MOSI
5	I	SPI SCK
6	I	SPI nCS-0 (M95080 EEPROM)
7	O	SPI MISO-0 (M95080 EEPROM)
8	I	SPI nCS-1 (TMP121 hőmérő)

Láb	Típus ¹	Funkció
9	O	SPI MISO-1 (TMP121 hőmérő)
10	OC	I ² C SDA
11	OC	I ² C SCL
12	OC	Hőmérséklet hiba kimenet (OS)
13		Nincs bekötve
14		Nincs bekötve
15		Nincs bekötve
16		Nincs bekötve

1-2. táblázat: A CON2 csatlakozó lábkiosztása (tesztpontok)

Láb	Funkció
1	SPI MOSI
2	SPI SCK
3	SPI nCS-0 (M95080 EEPROM)
4	SPI MISO-0 (M95080 EEPROM)
5	SPI nCS-1 (TMP121 hőmérő)

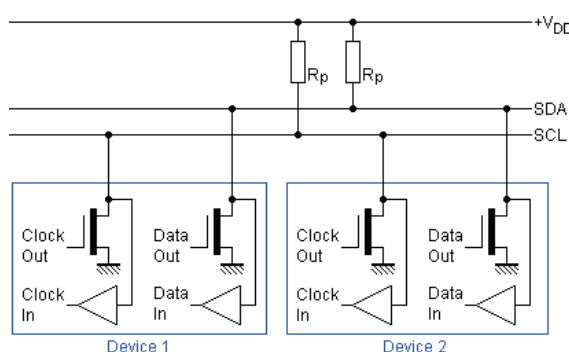
Láb	Funkció
6	SPI MISO-1 (TMP121 hőmérő)
7	I ² C SDA
8	I ² C SCL
9	Hőmérséklet hiba kimenet (OS)
10	GND (föld)

¹ PWR: tápellátás, I: bemenet, O: kimenet, OC: nyitott kollektoros meghajtás

Például: I (bemenet) esetén az FPGA számára az adott láb kimenet, a modul számára bemenet.

Az I²C (vagy másképpen I2C vagy IIC) egy igen gyakran használt áramkörök közötti kommunikációs interfész. Alkalmazásával akár több master és több slave egységet is összekapcsolhatunk. A master egység vezérli a buszt és biztosítja az órajelet. A slave eszköz a master egység kérésére válaszol és adatot fogad a master egység felől vagy küld a master egység felé. Az I²C busz nagy előnye, hogy a kommunikáció megvalósításához összesen két vezetékre van szükség, hátránya a limitált átviteli sebesség. A beágyazott rendszerekben leggyakrabban alkalmazott verzió az 1.0, amely 400 kHz-es maximális órajel frekvenciát engedélyez.

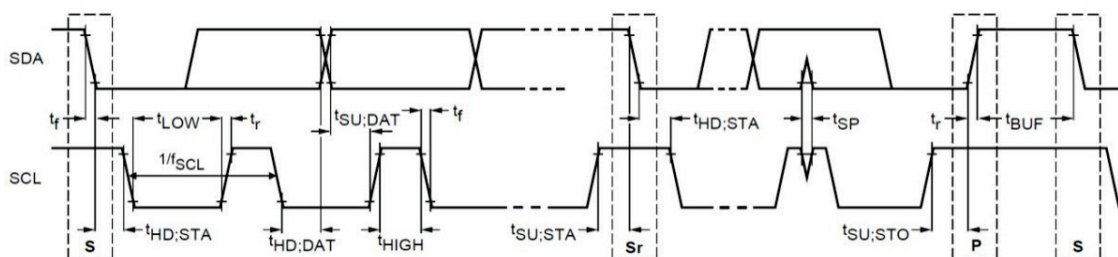
Az I²C busz fizikai interfésze két vezetékből áll, ezt szemlélteti a 2-1. ábra. Az SCL jel a szinkron működésű busz órajel vezetéke, az SDA jel pedig a kétirányú adatvonal. A szabvány szerint mindkét vezetéket nyitott kollektoros meghajtással kell vezérelni. Ez azt jelenti, hogy amennyiben a vezetékre logikai 0-át szeretnénk küldeni, a chipen belüli tranzisztort úgy vezéreljük, hogy az a vezetéket erősen nullába húzza le. Abban az esetben, ha a vezetékre logikai 1-et szeretnénk küldeni, a tranzisztort nem vezéreljük, így az R_p felhúzó ellenállás logikai magas szintet biztosít (feltéve, hogy a buszra kapcsolódó más eszközök közül semelyik nem húzza le a vezetéket logikai alacsony szintre).



2-1. ábra: Az I²C busz fizikai interfésze.

Mivel a logikai magas szintet a felhúzó ellenállás (általában 1 – 10 kΩ közötti érték) biztosítja, a csatlakoztatott eszközök és a busz kapacitása nagymértékben lassíthatja az átvitelt. A szabvány szerint a maximálisan 400 kHz frekvenciájú órajel használata esetén legfeljebb 400 pF kapacitás engedhető meg a buszon.

Az I²C buszon történő kommunikáció általános idődiagramja a 2-2. ábrán látható, az időzítési paramétereket a 2-1. táblázat tartalmazza.



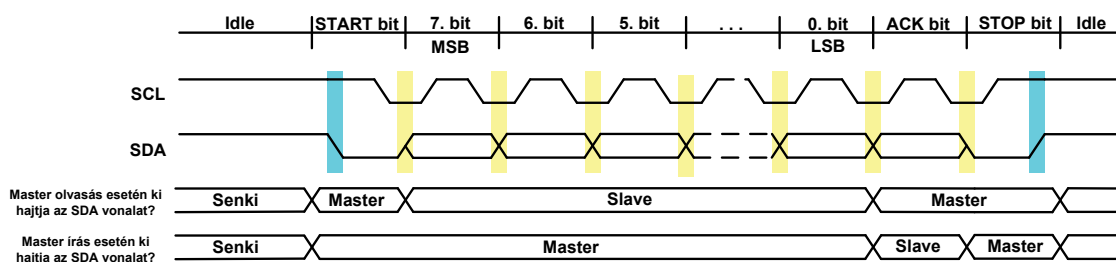
2-2. ábra: Az I²C busz időzítése.

2-1. táblázat: Az I²C busz időzítési paraméterei.

Paraméter	Szimbólum	Normál mód		Fast mód		Egység
		Min.	Max.	Min.	Max.	
SCL órajel frekvencia	f _{SCL}	0	100	0	400	kHz
(Ismételt) START feltétel tartási ideje	t _{HD;STA}	4,0	-	0,6	-	μs
SCL órajel alacsony szintjének ideje	t _{LOW}	4,7	-	1,3	-	μs
SCL órajel magas szintjének ideje	t _{HIGH}	4,0	-	0,6	-	μs
Ismételt START feltétel előkészítési ideje	t _{SU;STA}	4,7	-	0,6	-	μs
Adat tartási idő	t _{HD;DAT}	0	3,45	0	0,9	μs
Adat előkészítési idő	t _{SU;DAT}	250	-	100	-	ns
SDA/SCL felfutási idő	t _r	-	1000	20	300	ns
SDA/SCL lefutási idő	t _f	-	300	20	300	ns
STOP feltétel előkészítési ideje	t _{SU;STO}	4,0	-	0,6	-	μs
Szabad busz idő STOP és START feltétel között	t _{BUF}	4,7	-	1,3	-	μs
A tűskék megengedett pulzusszélessége	t _{SP}	-	-	0	50	ns

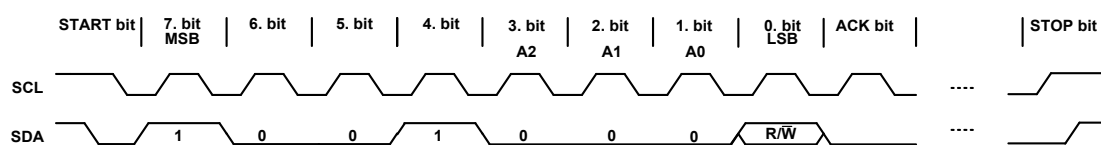
Az I²C adatátvitel az alábbi elemi átviteli blokkból épül fel, melyet a 2-3. ábra is szemléltet:

1. Nem használt állapotban (Idle) a busz egyik vezetéke sincs meghajtva, tehát logikai magas szinten vannak az R_p felhúzó ellenállások hatására.
2. Ha a master adatot szeretne küldeni, akkor egy speciális START feltételt (START bitet) ad ki: az SCL magas állapota alatt az SDA vonalon lefutó élet generál.
3. Ezek után következnek az adatbitek. Az SCL vonalon a master egy órajelet generál, az SDA vonalra pedig a megfelelő bit kerül, legelőször a legmagasabb helyiértékű bit (MSB). Fontos megjegyezni, hogy az SDA vezetéken jelváltás csak az SCL alacsony állapota alatt (ez az ábrán sárga színnel van jelölve) engedhető meg az adatbitek elküldése esetén.
 - a. Ha a master adatot ír, akkor az SDA vonalat a master hajtja meg.
 - b. Ha a master adatot olvas, akkor az SDA vonalat slave hajtja meg.
4. A nyolcadik adatbit átvitele után egy kilencedik, nyugtázó (acknowledge, ACK) bit átvitele történik meg, ehhez a master még egy órajelet generál az SCL vonalon.
 - a. Ha a master adatot ír, akkor az ACK alatt az SDA vonalat a slave hajtja meg
 - b. Ha a master adatot olvas, akkor az ACK alatt az SDA vonalat a master hajtja meg.
5. Amennyiben több mint egy bájtot viszünk át, a master további 8+1 órajelet generál, ebben az esetben a folyamat ismétlődik a 3. ponttól kezdődően. Ellenkező esetben az adatátvitel végén a master egy speciális STOP feltételt (STOP bitet) ad ki: az SCL magas állapota alatt az SDA vonalon felfutó élet generál. Ezután a busz nem használt (Idle) állapotba kerül.

2-3. ábra: Az I²C adatátvitel fázisai.

Általában elmondható, hogy az ACK bit értéke jelzi, hogy van-e szükség további bájt átvitelére. Ha az ACK bit értéke logikai 0, akkor az átvitel folytatódik, ha az ACK értéke logikai 1 (NACK) a megszólított eszköz nem válaszol, vagy nem akar több adatot átvinni.

A legelső átvitt bájt minden esetben egy speciális cím (address) bájt (2-4. ábra), amely segítségével megszólíthatunk egy slave eszközt és kijelölhetünk további kommunikációra. Így valósulhat meg, hogy egyetlen buszra akár több slave eszköz is felfűzhető. Fontos viszont, hogy minden slave eszköznek saját címmel kell rendelkeznie. Ahhoz, hogy azonos típusú eszközökből többet is használhassunk egyetlen buszon, a chipeket általában cím kiválasztó bemenetekkel is ellátják, így mindegyikhez külön-külön cím rendelhető hozzá.

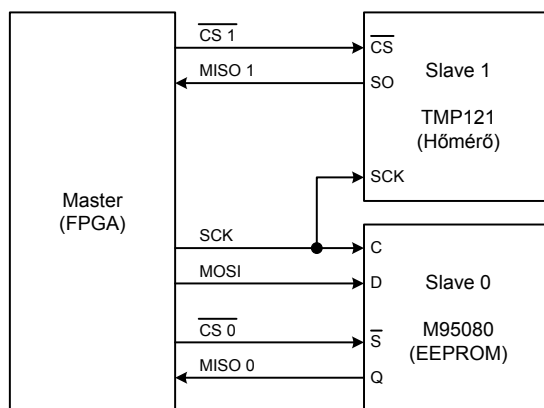


2-4. ábra: Az I²C cím bájt formátuma.

Ha a master írni szeretne a kiválasztott slave-re, akkor az R/W bit értéke 1, ha olvasni szeretne, akkor pedig az R/W bit értéke 0. A cím bájt küldésekor az ACK bitnek kitüntetett szerepe van. A slave eszköz a nyugtázással tudja jelezni, hogy a buszra csatlakozott és képes az adat továbbítására. Amennyiben senki sem húzta le a buszt az ACK fázisban, akkor az ACK bit értéke 1 lesz az R_p felhúzó ellenállás miatt, így egyértelműen kiderül, hogy a megcímezett eszköz nem válaszol.

3 Az SPI busz általános ismertetése

Az SPI (Serial Peripheral Interface) egy soros full-duplex adatátviteli interfész. Nagy előnye az I²C buszhoz képest, hogy csak egyirányú vonalakat használ, így akár több tíz MHz-es frekvencián is működhet. A buszra egy master és több slave eszköz kapcsolódhat. A modulon két slave periféria kapott helyet (3-1. ábra): az egyik egy TMP121 típusú hőmérő chip, a másik egy M95080 típusú EEPROM memória. A master funkciót az FPGA tölti be.



3-1. ábra: A modul SPI egységei.

Az SPI interfész az alábbi jelekből áll:

- SCK (Serial Clock): órajel, forrása mindig a master egység.
- MOSI (Master Output Slave Input): a master egység soros adatkimenete.
- MISO (Master Input Slave Output): a slave egység soros adatkimenete.
- nCS/SS (Chip Select / Slave Select): aktív alacsony kiválasztó jel, a master adja ki.

Az SPI szabvány szerint minden slave egység saját nCS kiválasztó jellel rendelkezik, de egymással megosztott MISO kimenetet használnak. A modulon ezzel szemben minden slave egység saját kiválasztó jellel (nCS0, nCS1) és saját adatkimenettel (MISO0, MISO1) rendelkezik.

A master és slave egységeknek meg kell állapodnia, hogy hogyan értelmezik az órajelet. Ennek megfelelően négyféle SPI adatátviteli mód lehetséges, amelyet az úgynevezett CPOL (órajel polaritás) és CPHA (órajel fázis) bitekkel szoktak jellemezni:

- Ha a CPOL üzemmód bit 0, akkor az SCK órajel alacsony szintű az inaktív állapotban.
 - Ha a CPHA üzemmód bit 0, akkor az adatok mintavételezése az SCK órajel felfutó élére történik, az adatok küldése pedig az SCK órajel lefutó élére.
 - Ha a CPHA üzemmód bit 1, akkor az adatok mintavételezése az SCK órajel lefutó élére történik, az adatok küldése pedig az SCK órajel felfutó élére.
- Ha a CPOL üzemmód bit 1, akkor az SCK vonal magas szintű az inaktív állapotban.
 - Ha a CPHA üzemmód bit 0, akkor az adatok mintavételezése az SCK órajel lefutó élére történik, az adatok küldése pedig az SCK órajel felfutó élére.
 - Ha a CPHA üzemmód bit 1, akkor az adatok mintavételezése az SCK órajel felfutó élére történik, az adatok küldése pedig az SCK órajel lefutó élére.

4 LM75 – I²C buszos hőmérő

Az LM75 egy igen széles körben elterjedt I²C buszra csatlakoztatható, konfigurálható hőmérő chip (adatlapja letölthető a <http://datasheets.maxim-ic.com/en/ds/LM75.pdf> címről). Ebben az alfejezetben az LM75 áramkör elsődleges funkcióját, a hőmérséklet kiolvasását fogjuk bemutatni.

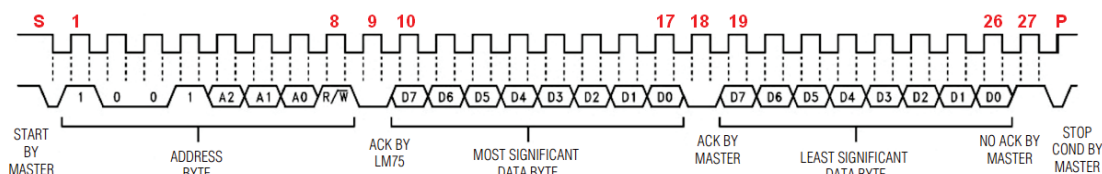
A jelen kialakításban egyetlen master és egyetlen slave eszköz kapcsolódik össze, így a master (FPGA) az SCL vonalat akár normál kimenetként is használhatja. Az SDA adatvezeték viszont minden esetben kétirányú, nyitott kollektoros meghajtást igényel. A modulon az LM75 A2-A0 címkiválasztó bemenetei földre vannak kötve, így az eszköz címe az I²C buszon 0x48, tehát olvasáskor a cím bájt értéke 0x91, íráskor pedig 0x90.

A mért hőmérséklet 0.5 °C-os felbontással (± 2 °C pontossággal) egy belső 16 bites regiszterben tárolódik az alábbiak szerint: a felső 9 biten található a tárolt hőmérséklet 0.5 °C-os felbontással, kettes komplement számábrázolási formátumban, a regiszter alsó 7 bite pedig nem használt (vegyük észre, hogy a felső bájt egyben a mért hőmérséklet érték 1 °C-os felbontással). Az adatbitek értelmezését és három minta hőmérséklet értéket a 4-1. táblázat tartalmazza. Az LM75 aktuális hőmérséklet regiszterének - bekapcsolás utáni - I²C buszon

történő olvasási idődiagramja a 4-1. ábrán látható, az adatátvitel egyes fázisait részletesen a 4-2. táblázat tartalmazza.

4-1. táblázat: Az LM75 IC által használt számbázis.

	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
	Egészrész								Tört-rész	Don't care						
+25.5 °C	0	0	0	1	1	0	0	1	1	X	X	X	X	X	X	X
+25 °C	0	0	0	1	1	0	0	1	0	X	X	X	X	X	X	X
-25 °C	1	1	1	0	0	1	1	1	0	X	X	X	X	X	X	X



4-1. ábra: Az aktuális hőmérséklet regiszter olvasása (bekapcsolás után).

4-2. táblázat: Az aktuális hőmérséklet regiszter olvasásának fázisai.

Órajel sorszáma	Az SDA vonalat meghajtja	Leírás
Start (S)	master (FPGA)	START feltétel generálása,
1-8	master (FPGA)	Cím bájtt kiadása: b10010001 = 0x91 (olvasás).
9	slave (LM75)	Az LM75 ACK = 0 bittel jelzi, hogy készen áll az átvitelre.
10-17	slave (LM75)	A hőmérséklet adat felső bájttjának átvitele.
18	master (FPGA)	A master ACK=0 bittel jelzi, hogy további bájttot szeretne olvasni.
19-26	slave (LM75)	A hőmérséklet adat alsó bájttjának átvitele.
27	master (FPGA)	A master ACK=1 bittel jelzi, hogy nem olvas további bájttokat.
Stop (P)	master (FPGA)	Az átvitelt a master a STOP feltétel generálásával zárja le.

A korábban ismertetett aktuális hőmérséklet regiszteren túl az eszköz egyéb funkciókkal is rendelkezik. A chip összesen 4 regisztert tartalmaz, amelyet a 4-3. táblázat foglal össze. Az eszköz konfigurációs regiszterében található bitek funkcióit a 4-4. táblázat tartalmazza. Fontos megjegyezni, hogy a 16 bites regiszterek olvasásakor (pl. aktuális hőmérséklet regiszter) a master-nek minden esetben kötelező mind a két bájttot kiolvasnia, különben az LM75 blokkolja az I²C buszt. (Az első bájtt olvasása után, az LM75 az SDA vonalat alacsonyban tartja, ezáltal lehetetlenné teszi a STOP bit generálását. Amikor megtörtént a második bájtt kiolvasása, az LM75 engedi a buszciklust lezárni.)

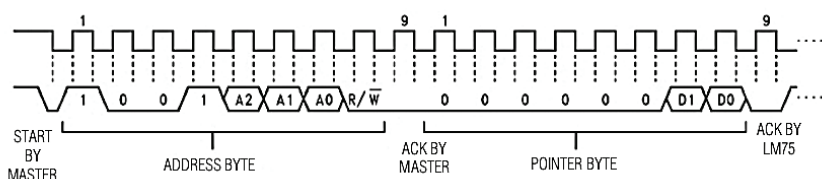
4-3. táblázat: Az LM75 áramkör regiszterei.

Regiszter cím	R/W	Szélesség	Név	Reset érték
b00	csak olvasható	16 bit	aktuális hőmérséklet regiszter	-
b01	írható/olvasható	8 bit	konfigurációs regiszter	0
b10	írható/olvasható	16 bit	T _{HYST} regiszter	75 °C
b11	írható/olvasható	16 bit	T _{OS} regiszter	80 °C

4-4. táblázat: A konfigurációs regiszter bitjeinek funkciói.

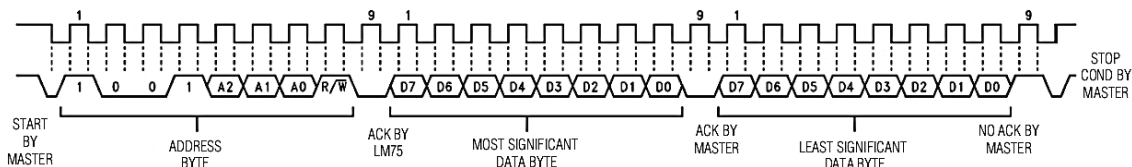
Bit	Név	Leírás
Bit 0	Készlet	0: normál működés 1: az LM75 készlet állapotba kerül
Bit 1	Cmp/Int	0: komparátor üzemmód 1: megszakítás üzemmód
Bit 2	OS polaritás	0: az OS kimenet aktív alacsony 1: az OS kimenet aktív magas
Bit 4-3	Megengedett hibák száma	A hőmérséklet hibák száma, mielőtt azt az O.S. kimenet jelezné: 00: jelzés az első hibánál (azonnali jelzés) 01: jelzés a második hibánál 10: jelzés a negyedik hibánál 11: jelzés a hatodik hibánál
Bit 7-5	Nem használt	Értéke kötelezően 0.

Azt, hogy a négy belső regiszter közül pontosan melyiket érjük el, egy belső kétbites pointer (mutató) regiszter tartalmazza. A pointer regiszter csak írható, reset után az értéke 0 (tehát az aktuális hőmérséklet regiszter kerül kiválasztásra). Ahhoz, hogy a többi regisztert is elérhessük, szükséges a pointer regiszter beállítása, melynek értékét az írási művelet során a cím bájtot követő bájtban kell megadni (4-2. ábra). A pointer regiszter a beállított értékét kikapcsolásig megőrzi, tehát ha ugyanazt a belső regisztert többször is olvassuk, nem szükséges előtte minden alkalommal a regiszter címet beállítani.

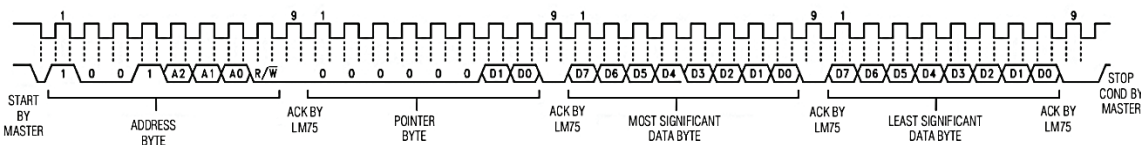


4-2. ábra: A pointer regiszter beállítása.

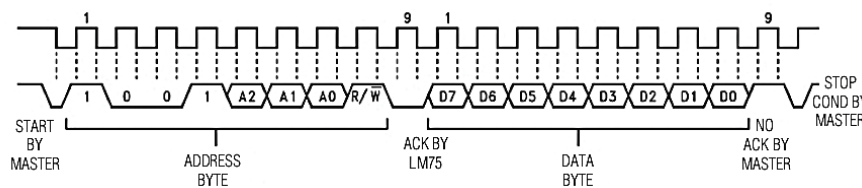
A 16 bites regiszterek olvasási idődiagramja a 4-3. ábrán látható, az írási idődiagramot a 4-4. ábra szemlélteti. A 8 bites konfigurációs regiszter olvasási idődiagramja a 4-5. ábrán látható, a regiszter írásának idődiagramját a 4-6. ábra mutatja. A regiszterek írása esetén a cím bájt kiadása után következik a pointer regiszter értéke, majd pedig a regiszter méretétől függően a beírandó 1 vagy 2 bájtnyi adat. Olvasás esetén a cím bájt kiadása után közvetlenül az 1 vagy 2 bájtos adat következik.



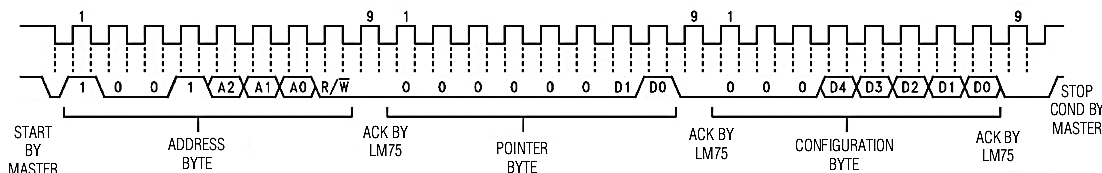
4-3. ábra: 16 bites regiszter olvasása (korábban beállított pointer regiszter mellett).



4-4. ábra: 16 bites regiszter írása.

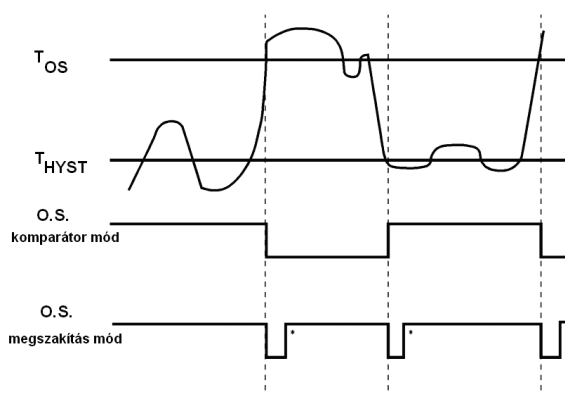


4-5. ábra: A 8 bites konfigurációs regiszter olvasása.



4-6. ábra: A 8 bites konfigurációs regiszter írása.

A hőmérséklet hiba kimenet (OS) komparátor módban felhasználható, mint egy egyszerű termosztát (pl. egy ventilátor vezérlésére). A felhasználó a T_{HYST} és a T_{OS} regiszterekkel tudja beállítani a kívánt kapcsolási hőmérsékletet. Mivel két hőmérsékletet kell definiálni, tetszőleges hiszterézis valósítható meg. Megszakítás módban az OS kimenetet egy mikroprocesszor megszakítás lábához lehet kötni. Az aktív megszakítási állapot bármely regiszter kiolvasásával feloldható. Az OS kimenet kétféle üzemmódját a 4-7. ábra szemlélteti.



4-7. ábra: Hőmérséklet hiba kimenet használata.

5 TMP121 – SPI buszos hőmérő

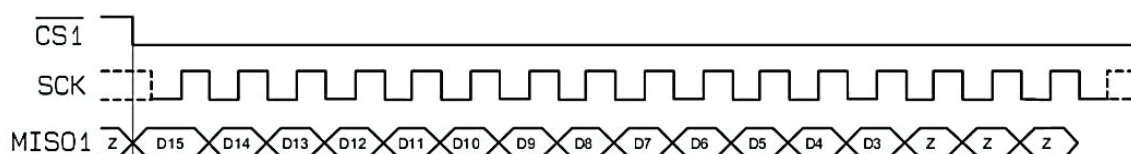
A TMP121 egy meglehetősen egyszerű integrált áramkör, amely csak a mért hőmérsékletet továbbítja a master felé a MISO1 vonalon át. Mivel a chipen semmilyen beállítási lehetőség nincs, ezért nincs is MOSI bemenete. Az TMP121 hőmérő áramkör adatlapja letölthető a <http://www.ti.com/product/tmp121> címről.

A mért hőmérséklet 0.0625 °C-os felbontással (± 1.5 °C pontossággal) egy belső 16 bites regiszterben tárolódik. A felső 13 biten található a tárolt hőmérséklet 0.0625 °C-os felbontással, kettes komplement számábrázolási formátumban, míg a regiszter alsó 3 bitje nem használt. A 13 biten tárolt hőmérséklet felső 9 bitje a mért hőmérséklet egészrésze, míg az alsó 4 bit a mért hőmérséklet törtrésze. Az adatbitek értelmezését és három minta hőmérséklet értéket az 5-1. táblázat tartalmazza.

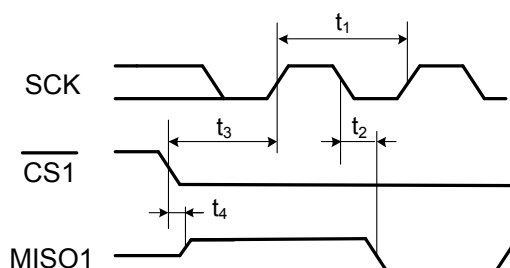
5-1. táblázat: A TMP121 chip által használt számábrázolás.

	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
	Egészrész									Törtrész				Don't care		
+25 °C	0	0	0	0	1	1	0	0	1	0	0	0	0	X	X	X
0.0625 °C	0	0	0	0	0	0	0	0	0	0	0	0	1	X	X	X
0 °C	0	0	0	0	0	0	0	0	0	0	0	0	0	X	X	X
-0.0625 °C	1	1	1	1	1	1	1	1	1	1	1	1	1	X	X	X
-25 °C	1	1	1	1	0	0	1	1	1	0	0	0	0	X	X	X

A TMP121 áramkörből a mért hőmérséklet értéket SPI interfészen lehet kiolvasni. Az olvasás idődiagramját az 5-1. ábra mutatja. Az SPI kiválasztó jel ($\overline{\text{CS1}}$) aktív ideje alatt (logikai alacsony) 16 órajel (SCK) ütemben elküldésre kerül a 16 adatbit az IC soros adatkimenetén (MISO1). Az IC adatlapja szerint (és az ábrából láthatóan) az SPI kommunikáció során a TMP121 az SCK órajel lefutó élére shifteli ki a 16 bitet. Az áramkör SPI interfészének időzítési paramétereit az 5-2. ábra és az 5-2. táblázat mutatja.



5-1. ábra: A TMP121 adatkimenete.



5-2. ábra: A TMP121 SPI interfészének időzítése.

5-2. táblázat: A TMP121 SPI interfészének időzítési paraméterei.

Szimbólum	Paraméter	Min.	Max.	Egység
t_1	SCK órajel periódusideje	100	-	ns
t_2	MISO1 késleltetési ideje az SCK lefutó éle után	-	30	ns
t_3	nCS1 előkészítési idő	40	-	ns
t_4	MISO1 késleltetési ideje az nCS1 lefutó éle után	-	30	ns

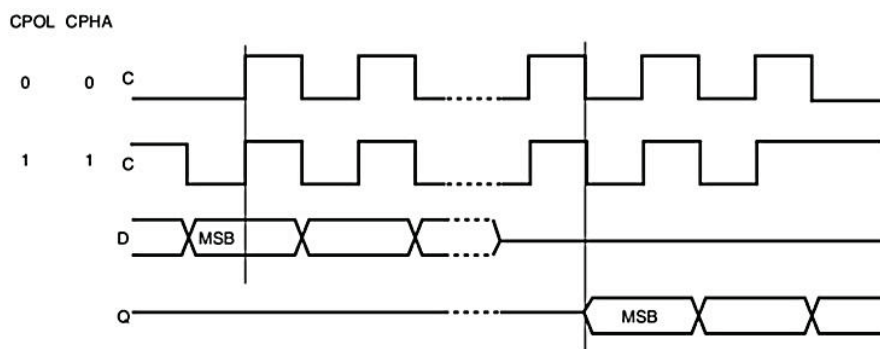
Az IC a hőmérséklet mérését akkor végzi el, amikor a chip nincs kiválasztva, azaz az SPI kiválasztó jel (nCS1) magas szintű. Egy konverzió időtartama 320 ms, az aktuális hőmérséklet kiolvasásához garantálnunk kell, hogy az nCS1 jel legalább ennyi ideig magas állapotú legyen. Abban az esetben, ha a kiválasztó jel huzamosabb ideig magas állapotban van, a chip 0,5 s periódusidővel frissíti a hőmérsékletet tároló belső regiszter értékét. A kiválasztó jel lefutó éle az esetlegesen futó aktuális mérési folyamat félbeszakad és a regiszterbe utoljára beírt érték olvasható ki.

6 M95080 – SPI buszos EEPROM

A modulon található másik SPI buszra csatlakozó eszköz az M95080 8 kbites (1 kBájtos) EEPROM memória, amely valódi kétirányú SPI kommunikációt valósít meg. Az eszköz adatlapja letölthető a gyártó honlapjáról: <http://www.st.com>.

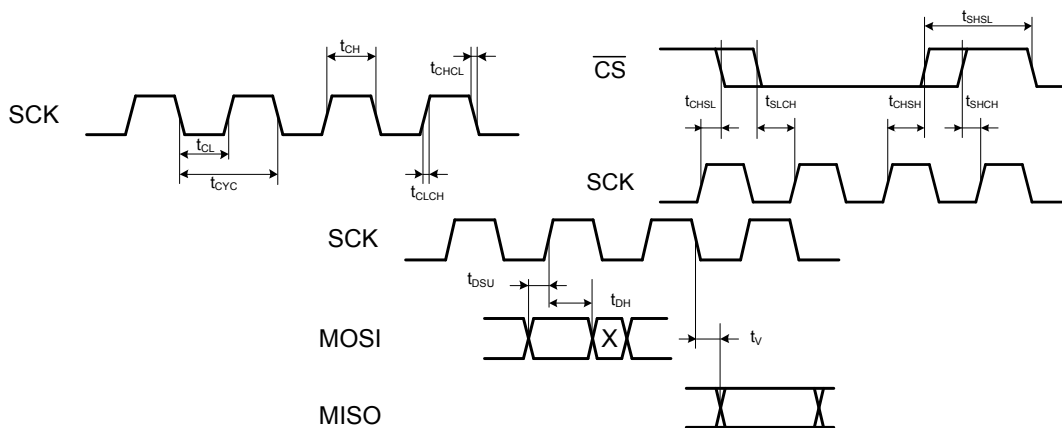
6.1 Az EEPROM SPI interfésze

Az M95080 EEPROM memória a CPOL, CPHA = 00 vagy a CPOL, CPHA = 11 üzemmódokat támogatja (6-1. ábra). Tehát a MOSI (D) vonalon a bejövő adatokat mindig az órajel felfutó élénél mintavételezi és a MISO0 (Q) kimenetre pedig az órajel lefutó élének hatására helyez érvényes adatot. Az SPI interfészen az adatok kishíftelése mindig a legnagyobb helyiértékű bittel kezdődik (MSB first).



6-1. ábra: Az órajel időzítési viszonyai

Az M95080 áramkör SPI interfészének időzítési paramétereit a 6-2. ábra és a 6-1. táblázat szemlélteti.



6-2. ábra: Az M95080 EEPROM SPI interfészének idődiagramja.

6-1. táblázat: Az M95080 EEPROM SPI interfészének időzítési paraméterei.

Szimbólum	Paraméter	Min.	Max.	Egység
t_{CYC}	SCK órajel periódusideje	200	-	ns
t_{CL} / t_{CH}	SCK órajel alacsony/magas szintjének ideje	90	-	ns
t_{CLCH} / t_{CHCL}	SCK órajel felfutási/lefutási ideje	-	2	μs
t_{SLCH} / t_{CHSH}	nCS előkészítési idő	90	-	ns
t_{CHSL} / t_{SHCH}	nCS tartási idő	90	-	ns
t_{SHSL}	nCS magas (inaktív) szintjének ideje	100	-	ns
t_{DSU}	Adatbemenet (MOSI) előkészítési idő	90	-	ns
t_{DH}	Adatbemenet (MOSI) tartási idő	90	-	ns
t_V	Adatkimenet (MISO) késleltetési idő	-	100	ns

6.2 Az EEPROM státusz regisztere

Az EEPROM státusz regiszterében található bitek értelmezését a 6-2. táblázat mutatja. A státusz regiszter írható bitjei (SRWD, BP1 és BP0) az értéküket a tápfeszültség kikapcsolása után is megőrzik.

6-2. táblázat: Az EEPROM státusz regiszterének bitjei.

Bit	Név	R/W	Funkció
Bit 0	WIP	R	Write In Progress – írás folyamatban Mivel az EEPROM írása 5 ms időt igényel, ezért a felhasználó ezen bit figyelésével győződhet meg arról, hogy az írási ciklus már befejeződött-e vagy sem. 0: nincs aktív írási folyamat 1: írási művelet folyamatban
Bit 1	WEL	R	Write Enable Latch – írás Engedélyezés Latch Minden írás típusú utasítás (WRITE, WRSR) kiadása előtt a WEL bitet egybe kell állítani a WREN utasítás kiadásával. Az alábbi események után a WEL bit ismét alapállapotba kerül: - Bekapcsolás - WRDI utasítás - WRSR utasítás befejezése - WRITE utasítás befejezése 0: EEPROM írás tiltva (alapállapot) 1: EEPROM írás engedélyezve

Bit 2-3	BP0/1	R/W	Az EEPROM memória egy része írásvédetté tehető ezen bitek segítségével. - BP[1:0] = 00 esetén nincs írásvédelem - BP[1:0] = 01 esetén a 0x300 – 0x3FF címtartomány írásvédett - BP[1:0] = 10 esetén a 0x200 – 0x3FF címtartomány írásvédett - BP[1:0] = 11 esetén a teljes memória írásvédett
Bit 4-6		R	Nem használt
Bit 7	SRWD	R/W	Status Register Write Disable – Státusz Regiszter Írás Tiltva Abban az esetben, ha az EEPROM chip nWE lába alacsony és az SRWD bit értéke egy, a státusz regiszter írhatósága letiltódik (hardveres védelem). Ezáltal lehet biztosítani a BP0 és BP1 bitek védelmét. Mivel a modulon az nWE láb fixen tápfeszültségre van kötve, ezért a hardveres védelem nem használható.

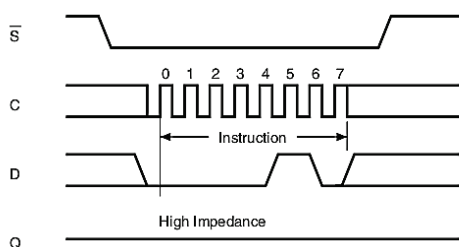
6.3 Az EEPROM parancskészlete

Az M95080 EEPROM memóriával parancsok segítségével kommunikálhatunk, amelyeket az SPI buszon keresztül továbbíthatunk az eszköznek. A 6-3. táblázat foglalja össze a memóriának elküldhető parancsokat.

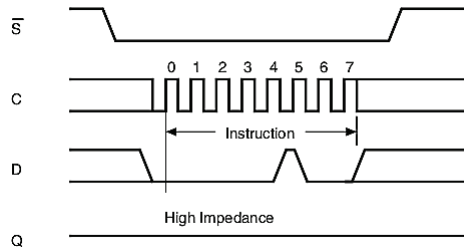
6-3. táblázat: Az M95080 EEPROM parancskészlete.

Parancs	Bináris	Hex	Funkció
WREN	b0000 0110	0x06	Írás engedélyezés
WRDI	b0000 0100	0x04	Írás tiltás
RDSR	b0000 0101	0x05	Státusz regiszter olvasás
WRSR	b0000 0001	0x01	Státusz regiszter írás
READ	b0000 0011	0x03	Memória olvasás
WRITE	b0000 0010	0x02	Memória írás

Minden írás típusú utasítást (WRITE, WRSR) meg kell előznie egy WREN (írás engedélyezés) utasításnak (6-3. ábra), amely hatására a státusz regiszter WEL bitjének értéke 1 lesz, ezáltal engedélyeződnek a WRITE és a WRSR utasítások. A WRDI utasítás (6-4. ábra) nem nagy gyakorlati jelentőséggel bír, hiszen az írási ciklusok végén a WEL bit ismét törlődik, tehát a memória írásvédetté válik.



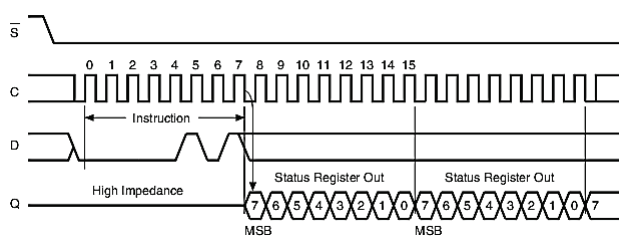
6-3. ábra: A WREN utasítás idődiagramja.



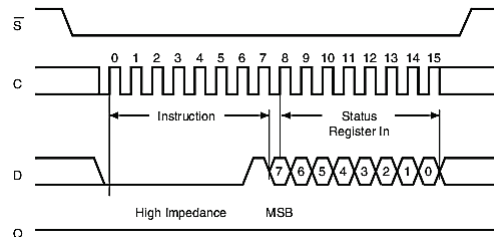
6-4. ábra: A WRDI utasítás idődiagramja.

Mivel az írási folyamat időtartama 5 ms, a státusz regiszter periodikus olvasásával és a WIP bit periodikus lekérdezésével vizsgálhatjuk meg, hogy az írás befejeződött-e. A státusz regiszter kiolvasása az RDSR utasítás kiadásával lehetséges (6-5. ábra). Az ábrán jól látható, hogy amennyiben a master egység további órajeleket biztosít, a státusz regiszter többször is kiolvasható egyetlen buszciklusban.

A WRSR parancs segítségével (6-6. ábra) írhatjuk a státusz regiszter SRWD, BP1 és BP0 bitjeit. Használata előtt ne felejtjük el kiadni a WREN parancsot. A státusz regiszter írásának időtartama 5 ms, az írási folyamat végét a WIP bit értékének vizsgálatával állapíthatjuk meg.

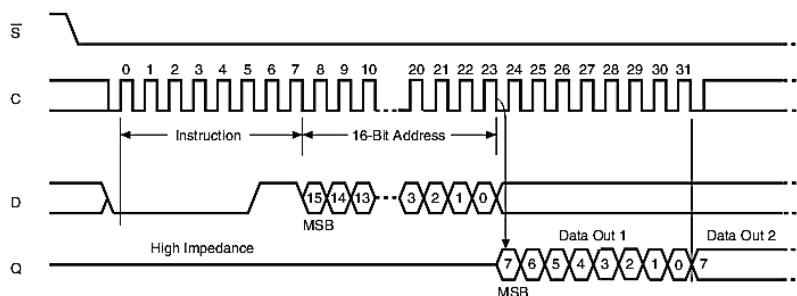


6-5. ábra: Az RDSR utasítás idődiagramja.



6-6. ábra: A WRSR utasítás idődiagramja.

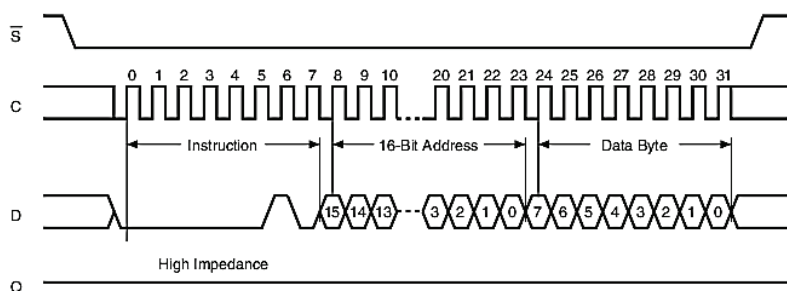
A 6-7. ábrán az EEPROM olvasási ciklusa látható. A READ parancs kishiftelése után 16 biten továbbítjuk az olvasni kívánt memóriacella címét (az 1024 byte megcímezéséhez 10 bitre van szükség, a felső 6 címbit értéke nincs figyelembe véve). Ezek után a következő órajel periódustól kezdve az EEPROM már továbbítja a kért adatot a MISO0 (Q) soros kimeneten keresztül. Amennyiben a master egység további órajel pulzusokat is küld, az EEPROM a cím automatikus növelésével a következő memóriacella értékét is az adatvonalra helyezi. Az olvasási folyamat tetszőleges hosszan folytatható. Amennyiben a belső címszámláló eléri a végértéket, automatikusan túlcsoordul, és folytatja az olvasást a nullás címtől kezdődően. Fontos megjegyezni, hogy az írási művelet alatt READ parancs nem adható ki.



6-7. ábra: A READ utasítás idődiagramja.

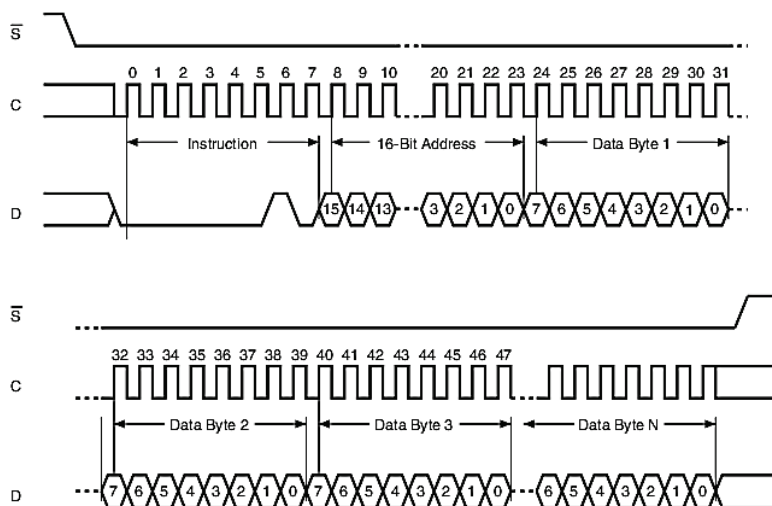
A WRITE parancs segítségével tudjuk a memória cellákat írni (6-8. ábra). A parancs kishiftelése után 16 biten továbbítjuk az írni kívánt memória cella címét, majd pedig a beírandó 8 bites adatot. Az írási folyamat időtartama 5 ms, melynek végét a WIP bit értékének vizsgálatával állapíthatjuk meg. Az adat beírása az alábbi esetekben nem történik meg:

- Ha a WEL bit értéke 0 (a WRITE utasítást meg kell előznie a WREN utasításnak).
- Ha a WIP bit értéke 1 (írási művelet van folyamatban).
- Ha BP1 és BP0 bitek által definiált módon írásvédett területre próbálunk írni.
- Ha az nCS alacsony szintje alatt kiadott órajel pulzusok száma nem osztható nyolccal.



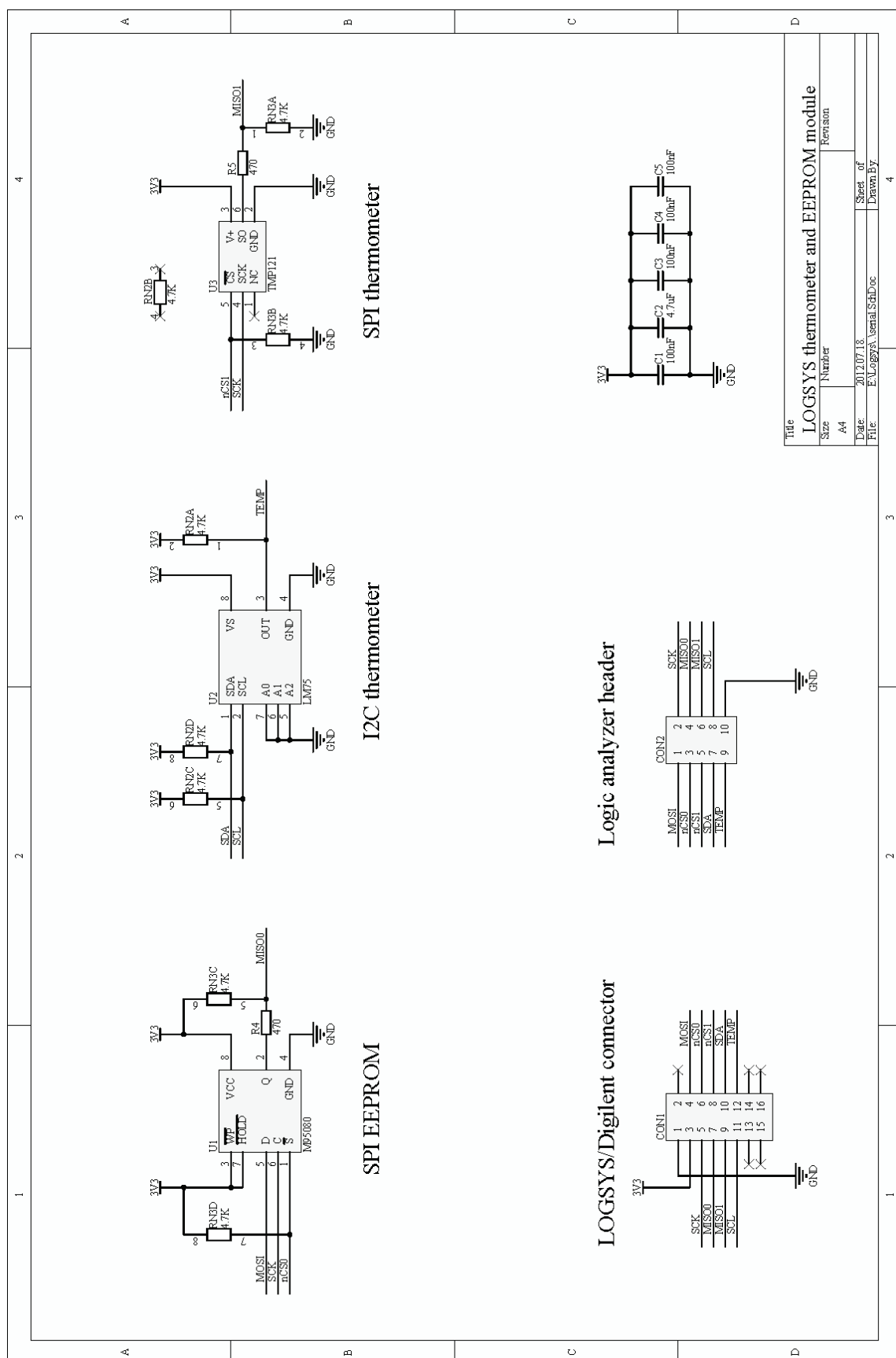
6-8. ábra: A WRITE utasítás idődiagramja.

A memóriacellák 32 bájtos lapokra vannak osztva. Amennyiben az írási buszciklus folytatódik, további bájtok írhatók be egyetlen WRITE utasítással (6-9. ábra). Fontos azonban megjegyezni, hogy az automatikusan növekvő címszámláló csak az alsó 6 címvezetőken számol (a laphatáron belül), tehát egyszerre maximum 32 bájt, azaz egyetlen lap írható be. Amennyiben az eszköznek küldött adatok laphatárt lépnek át, a címszámláló ugyanazon lap kezdetére és nem pedig a következő lapra fog mutatni.



6-9. ábra: Több bájt írása egyetlen WRITE utasítással.

7 A modul kapcsolási rajza



Változások a dokumentumban

Dátum	Verzió	Megjegyzés
2012. szeptember 16.	1.0	Az első kiadás.