

JÁRMŰFEDÉLZETI RENDSZEREK I.

6. előadás

Dr. Aradi Szilárd

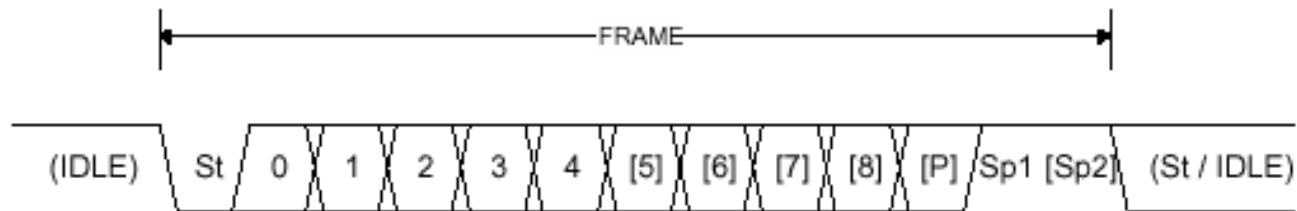
Dr. Bécsi Tamás



BME KÖZLEKEDÉSMÉRNÖKI ÉS JÁRMŰMÉRNÖKI KAR
32708-2/2017/INTFIN SZÁMÚ EMMI ÁLTAL TÁMOGATOTT TANANYAG

UART

- Kétvezetékes, aszinkron soros kommunikáció
- Kis távolság, alacsony átviteli sebesség
- Egyszerű, általános keretformátum
 - 5-9 adatbit
 - 1 startbit
 - 1 v. 2 stopbit
 - Opcionális paritásbit (páros vagy páratlan is lehet)



St Start bit, always low.

(n) Data bits (0 to 8).

P Parity bit. Can be odd or even.

Sp Stop bit, always high.

IDLE No transfers on the communication line (RxD or TxD). An IDLE line must be high.



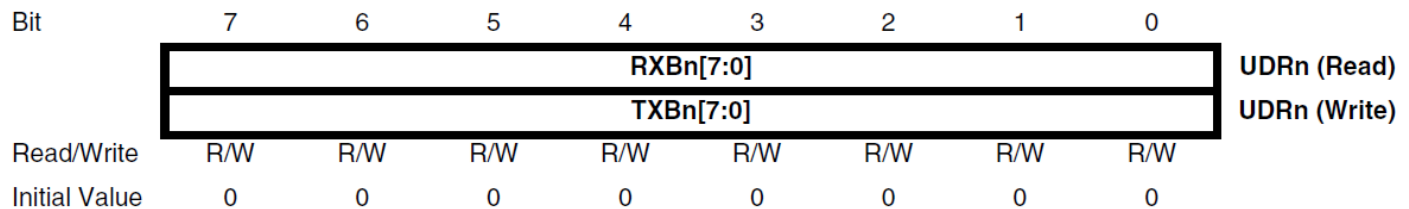
ATmega128 USART

- Universal Synchronous and Asynchronous serial Receiver and Transmitter (USART)
 - Szinkron működésre is képes (plusz egy órajel vezeték szükséges)
 - 2x sebességű működés támogatása aszinkron módban
 - 2 csatorna (USART0 és USART1)
 - USART0 csak aszinkron működésre képes
 - Az összes keretbeállítás támogatása (adatbitek, stopbitek száma)
 - Automatikus paritásbit generálás és ellenőrzés
 - Három megszakítás (küldés kész, küldő regiszter üres, fogadás kész)



USART regiszterek I.

- UDRn - USARTn I/O Data Register



- Küldő és fogadó regiszter

- Csak akkor írható, ha az UDREN regiszter egybe van állítva.
- Miután az adat be lett írva és a küldés engedélyezve van, az adatok betöltődnek a küldő shift regiszterbe (amennyiben az már üres) és megkezdődik a soros adatküldés.
- Fogadásnál a regiszteren keresztül a fogadó buffer - egy kétszintű FIFO memória kerül – kiolvasásra.



USART regiszterek II.

- UCSRnA – USART Control and Status Register A

Bit	7	6	5	4	3	2	1	0	
	RXCn	TXCn	UDREn	FEn	DORn	UPEn	U2Xn	MPCMn	UCSRnA
Read/Write	R	R/W	R	R	R	R	R/W	R/W	
Initial Value	0	0	1	0	0	0	0	0	

- USART Receive Complete (RXCn)

- A bit értéke 1, ha kiolvasatlan adat van a fogadó bufferben és 0, ha a buffer üres. Ha a fogadás letiltásra kerül, a buffer kiürítésre kerül és az RXCn nullázódik.

- USART Transmit Complete (TXCn)

- A bit értéke 1, ha a teljes adatkeret kiküldésre került a shift regiszteren keresztül és nincs új adat a küldő bufferben.

- USART Data Register Empty (UDREn)

- Ha a bit értéke 1, akkor írható új küldendő adat az UDRn regiszterbe.



USART regiszterek III.

- UCSRnA – USART Control and Status Register A

Bit	7	6	5	4	3	2	1	0	
	RXCn	TXCn	UDREn	FEn	DORn	UPEn	U2Xn	MPCMn	UCSRnA
Read/Write	R	R/W	R	R	R	R	R/W	R/W	
Initial Value	0	0	1	0	0	0	0	0	

- Frame Error (FEn)

- Kerethibát jelez, amely akkor áll elő, ha az első stopbit 0.

- Data OverRun (DORn)

- Akkor 1, ha a fogadó buffer (2 bájt) megtelt, egy további bájt várakozik a shift regiszterben és egy új startbit érkezett.

- Parity Error (UPEn)

- Akkor 1, ha paritáshibát érzékelt a fogadott adatban. A paritásellenőrzést engedélyezni kell hozzá (UPMn1=1).



USART regiszterek IV.

- UCSRnA – USART Control and Status Register A

Bit	7	6	5	4	3	2	1	0	
	RXCn	TXCn	UDREn	FEn	DORn	UPEn	U2Xn	MPCMn	UCSRnA
Read/Write	R	R/W	R	R	R	R	R/W	R/W	
Initial Value	0	0	1	0	0	0	0	0	

- Double the USART Transmission Speed (U2Xn)

- Ha 1, akkor dupla átviteli sebességgel működik a kommunikáció. Csak aszinkron módban működik, szinkron módban 0-ra kell állítani.

- Multi-Processor Communication Mode (MPCMn)

- Több processzoros mód beállítása. Ha 1, akkor a címet nem tartalmazó adatkeret eldobásra kerülnek.



USART regiszterek V.

- UCSRnB – USART Control and Status Register B

Bit	7	6	5	4	3	2	1	0	
	RXCIE_n	TXCIE_n	UDRIE_n	RXEN_n	TXEN_n	UCSZ_{n2}	RXB_{8n}	TXB_{8n}	UCSRnB
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- RX Complete Interrupt Enable (RXCIE_n)
 - Ha 1, akkor az adatfogadást jelző megszakítás engedélyezésre kerül.
- TX Complete Interrupt Enable (TXCIE_n)
 - Ha 1, akkor az adatküldés befejezését jelző megszakítás engedélyezésre kerül.
- USART Data Register Empty Interrupt Enable (UDRIE_n)
 - Ha 1, akkor az adatregiszter kiürülését jelző megszakítás engedélyezésre kerül.



USART regiszterek VI.

- UCSRnB – USART Control and Status Register B

Bit	7	6	5	4	3	2	1	0	
	RXCIE_n	TXCIE_n	UDRIE_n	RXEN_n	TXEN_n	UCSZ_{n2}	RXB_{8n}	TXB_{8n}	UCSRnB
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- Receiver Enable (RXEN_n)

- Ha 1, akkor engedélyezi az adatfogadást.

- Transmitter Enable (TXEN_n)

- Ha 1, akkor engedélyezi az adatküldést. Mindkettő felülvezérli a megfelelő portlábak működését.

- Character Size (UCSZ_{n2})

- Az UCSR_{nC} regiszterben lévő UCSZ_{n1:0} bitekkel együtt definiálja az adatbitek számát



USART regiszterek VII.

- UCSRnB – USART Control and Status Register B

Bit	7	6	5	4	3	2	1	0	
	RXCIE_n	TXCIE_n	UDRIE_n	RXEN_n	TXEN_n	UCSZ_{n2}	RXB_{8n}	TXB_{8n}	UCSRnB
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- Receive Data Bit 8 (RXB_{8n})

- A 9. fogadott adatbit, ha 9-bites mód van beállítva. Az UDR_n előtt kell kiolvani.

- Transmit Data Bit 8 (TXB_{8n})

- A 9. küldendő adatbit, ha 9-bites mód van beállítva. Az UDR_n előtt kell beírni.



USART regiszterek VIII.

- UCSRnC – USART Control and Status Register C

Bit	7	6	5	4	3	2	1	0	
	–	UMSELn	UPMn1	UPMn0	USBSn	UCSZn1	UCSZn0	UCPOLn	UCSRnC
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	1	1	0	

- USART Mode Select (UMSELn)

- 0 = aszinkron mód, 1 = szinkron mód

- Parity Mode (UPMn1:0)

- Paritás generálás és ellenőrzés.

00 = letiltva, 10 = páros paritás, 11 = páratlan paritás

- Stop Bit Select (USBSn)

- 0 = 1 stopbit, 1 = 2 stopbit



USART regiszterek IX.

- UCSRnC – USART Control and Status Register C

Bit	7	6	5	4	3	2	1	0	
	-	UMSELn	UPMn1	UPMn0	USBSn	UCSZn1	UCSZn0	UCPOLn	UCSRnC
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	1	1	0	

- Character Size (UCSZn1:0)

- Adatbitek száma. Hozzárakva az UCSZn2 bitet a B regiszterből.
000 = 5-bit, 001 = 6-bit, 010 = 7-bit, 011 = 8-bit, 111 = 9-bit

- Clock Polarity (UCPOLn)

- Csak szinkron módban használatos, aszinkron módban 0-án kell hagyni. A szinkron órajel vezeték (XCKn) polaritását adja meg.
- Ha 0, akkor a küldött adat az órajel felfutó élén változik meg, míg a fogadott adatot a lefutó élre kell mintavételezni. Ha 1, akkor fordítva működik.



USART regiszterek X.

- UBRRnL és UBRRnH – USART Baud Rate Registers

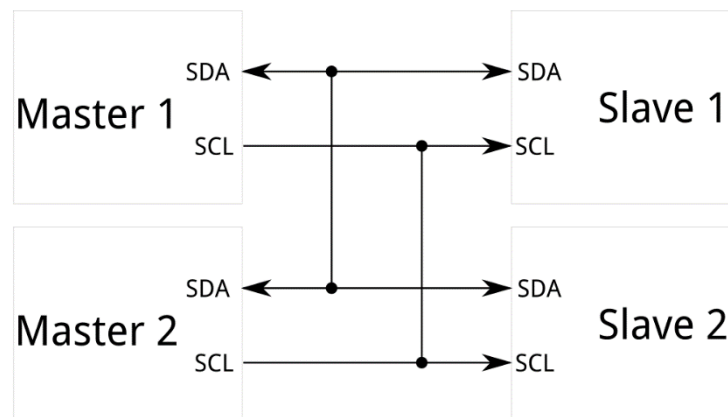
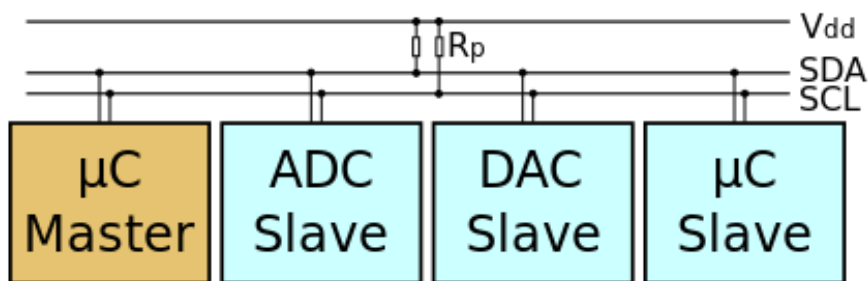
Bit	15	14	13	12	11	10	9	8	
	-	-	-	-	UBRRn[11:8]				UBRRnH
	UBRRn[7:0]								UBRRnL
	7	6	5	4	3	2	1	0	
Read/Write	R	R	R	R	R/W	R/W	R/W	R/W	
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	0	

- 12-bites regiszter
- UBRR regiszter számítás aszinkron 1x és 2x sebességnél: $UBRR = \frac{f_{osc}}{16BAUD} - 1$ $UBRR = \frac{f_{osc}}{8BAUD} - 1$
- Órajeltől függően nem lesz teljesen pontos a kívánt baud rate. 8-bites átvitelnél kb. 2% hiba megengedett.



I²C busz

- Az Inter-Integrated Circuit (I²C) buszt a Philips fejlesztette 1982-ben.
- Kiseb sebességű periféria IC-k és a mikrovezérlők közötti kommunikációra használható, rövid távolságon (nyomtatott áramkörön határain) belül
- Multi-master, master-slave, soros, szinkron buszrendszer
- 10 kHz (low-speed mode) és 100 kHz (standard mode) órajelfrekvencia
 - A legújabb verziójú eszközök ennél gyorsabbak is lehetnek:
 - 400 kHz Fast mode
 - 1 MHz Fast mode plus vagy Fm+
 - 3.4 MHz High Speed mode



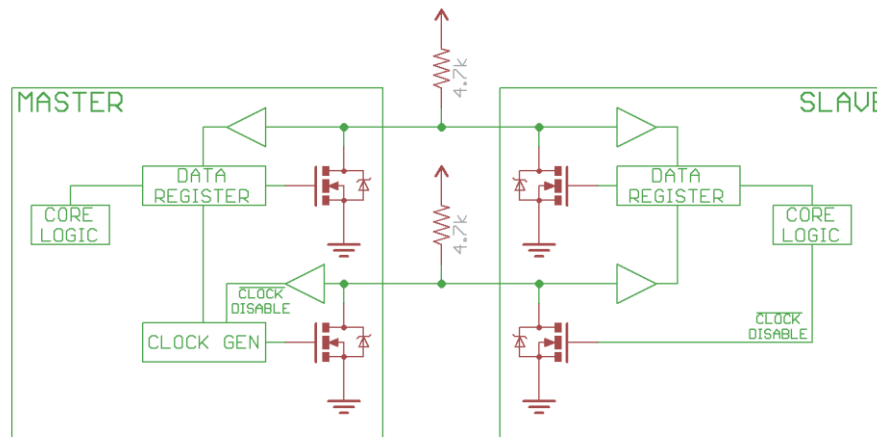
Forrás: [Wikipedia](https://en.wikipedia.org/wiki/I2C), [en:user:Cburnett](https://en.wikipedia.org/wiki/User:Cburnett), [CC BY-SA 3.0](https://creativecommons.org/licenses/by-sa/3.0/)

Forrás: [SparkFun Electronics](https://www.sparkfun.com/), [SFUPTOWNMAKER](https://www.sfuptownmaker.com/), [CC BY-NC-SA 3.0](https://creativecommons.org/licenses/by-nc-sa/3.0/)



I²C hardver

- Két darab kétirányú „open-drain” vezeték, ellenállásokkal tápfeszültségre felhúzva. Huzalozott ÉS kapcsolat, azaz az alacsony jelszint az erősebb.
- Serial Data Line (SDA) adatvezeték és Serial Clock Line (SCL) órajelvezeték
- Többféle feszültség szint is használható is a vezetékeken (pl.: 5V, 3.3V, 2.5V stb.)
 - 5V és 3,3V használható vegyesen, ilyenkor 3,3V-ra kell felhúzni a vezetékeket.
 - Egyéb esetben szintillesztő IC szükséges

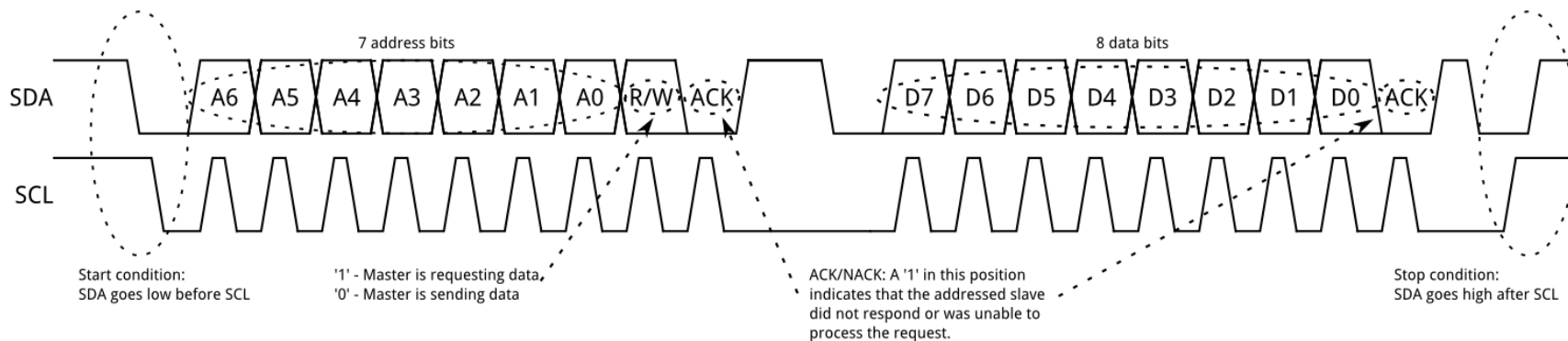


Forrás: [SparkFun Electronics](#), [SFUPTOWNMAKER](#), [CC BY-NC-SA 3.0](#)



I²C protokoll (master-slave) I.

- A kommunikáció üzenet (message) alapú, melyet start (start condition) és stop (stop condition) jelzések határolnak
- Egy üzenet több keretből (frame) áll
 - Egy címkeret (address frame)
 - Egy vagy több adatkeret (data frame)



Forrás: [SparkFun Electronics](#), [SFUPTOWNMAKER](#), [CC BY-NC-SA 3.0](#)

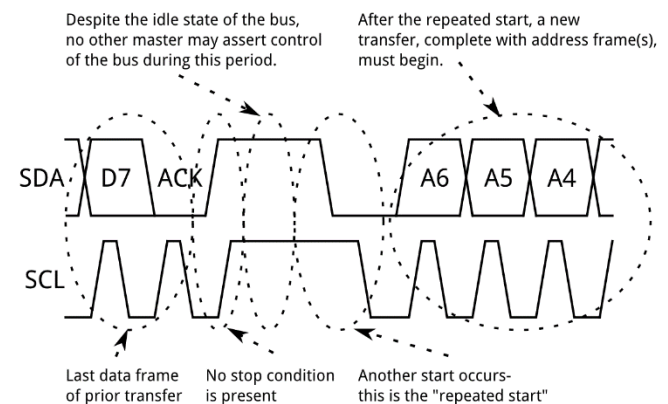
I²C protokoll (master-slave) II.

- Start Condition
 - A kommunikáció kezdetekor a master lehúzza az SDA-t, miközben az SCL magas szinten marad. (A busz alaphelyzetében mindkét vezeték magas jelszinten van.)
 - Erre kezdenek figyelni a slave-ek.
- Address Frame
 - Mindig ezzel kezdődik az üzenet.
 - 7-bites eszköz cím
 - 1-bites jelzés: írás vagy olvasási (R/W – 1/0) művelet
- Data Frame
 - Tetszőleges számú, 1 bájtos adat, master vagy slave tölti fel az R/W bittől függően
- ACK bit
 - Minden frame után a master elengedi az SDA-t (így az magas szintre kerül) és a megcímezett slave-nek kell alacsonyra húznia. Így jelzi, hogy fogadta az adatot.
- Stop Condition
 - Az üzenet végén a master lehúzza az SCL-t és az SDA-t az ACK után.
 - Felengedi az SCL-t, majd felengedi az SDA-t is.
 - Normál adatküldésnél az SDA nem változhat az SCL magas szintje mellett!



I²C protokoll (multi-master)

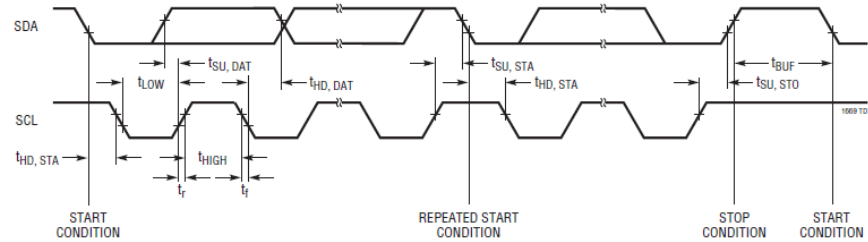
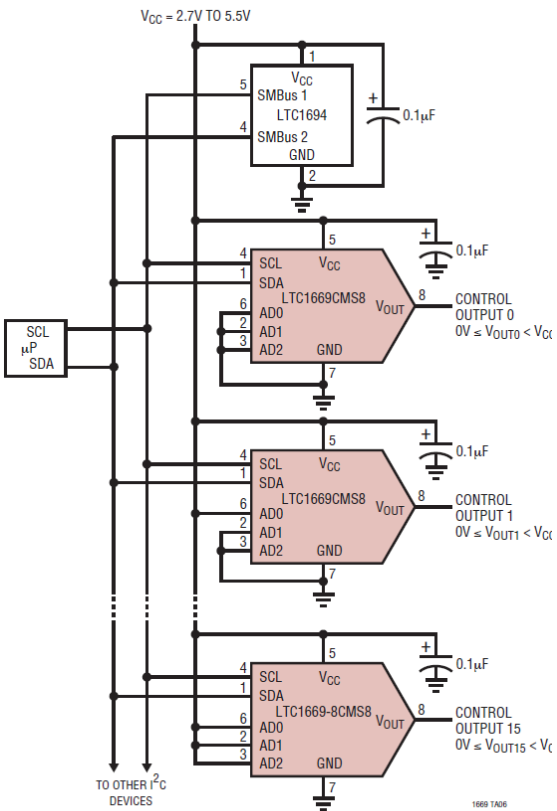
- A slave-ek működése ugyanolyan, mint master-slave esetben.
- A mestereknek viszont tudniuk kell a multi-master módot. Folyamatosan vissza kell olvasniuk a buszt, hogy a kiküldött adatuk jelent-e meg. Nem megfelelő mesterek alkalmazása hibához vezet!
- A mesterek figyelik a buszt, aki előbb húzza le az SDA-t azé a buszhasználat joga.
- Ha mégis egyszerre küldenek, akkor a huzalozott ÉS kapcsolat miatt egyszerre 1 és 0 bit estén az 1-es bit veszít. A vesztes master elengedi a buszt, majd később újra próbálkozik. Ezt a folyamatot hívják arbitrációnak.
- Ismételt Start Condition
 - Előfordulhat, hogy egy master több üzenet szeretne küldeni
 - Ekkor Stop Condition helyett Repeated Start Conditiont küld
 - Alacsony SCL mellett felengedik az SDA-t, majd az SCL-t és jön a Start Condition



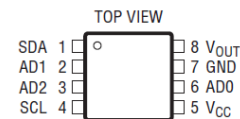
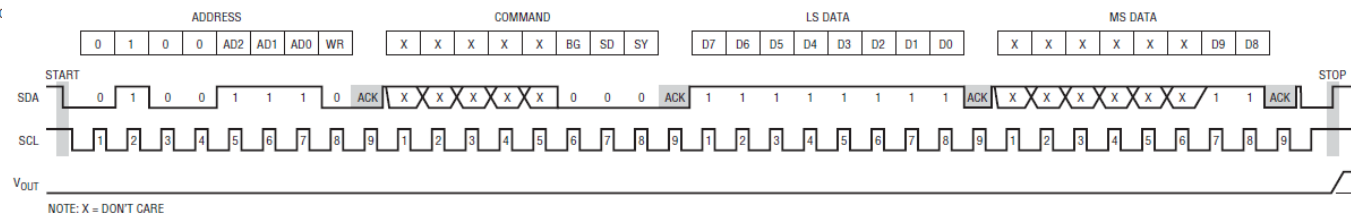
Forrás: [SparkFun Electronics](#), [SFUPTOWNMAKER](#), [CC BY-NC-SA 3.0](#)



I²C példa



Typical LTC1669 Input Waveform—Programming DAC Output for Full Scale (AD2 to AD0 Set High)



MS8 PACKAGE
8-LEAD PLASTIC MSOP

Atmel TWI (I²C)

- A Two-wire Serial Interface (TWI) az I²C megfelelője az Atmelnél.
- Támogatja a master és a slave működést is.
- Multi-master arbitrációs támogatás
- 7-bites címtartomány.
- Max. 400 kHz órajel.
- Címfelismerés ébresztési funkcióval alvó módból.



TWI regiszterek I.

- TWI Bit Rate Register (TWBR)

Bit	7	6	5	4	3	2	1	0									
	<table border="1"><tr><td>TWBR7</td><td>TWBR6</td><td>TWBR5</td><td>TWBR4</td><td>TWBR3</td><td>TWBR2</td><td>TWBR1</td><td>TWBR0</td></tr></table>								TWBR7	TWBR6	TWBR5	TWBR4	TWBR3	TWBR2	TWBR1	TWBR0	TWBR
TWBR7	TWBR6	TWBR5	TWBR4	TWBR3	TWBR2	TWBR1	TWBR0										
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W									
Initial Value	0	0	0	0	0	0	0	0									

– Órajel (SCL) frekvencia beállítása

$$\text{SCL frequency} = \frac{\text{CPU Clock frequency}}{16 + 2(\text{TWBR}) \cdot 4^{\text{TWPS}}}$$

– TWPS: előosztó bitek értéke a TWSR regiszterben



TWI regiszterek II.

- TWI Control Register (TWBR)

Bit	7	6	5	4	3	2	1	0	
	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	–	TWIE	TWCR
Read/Write	R/W	R/W	R/W	R/W	R	R/W	R	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- TWI Interrupt Flag (TWINT)

- Beáll 1-be, ha a TWI aktuális feladata véget ért. Magától nem törlődik, szoftveresen kell törölni 1-et írva bele. Törlés után a TWI megkezdi a következő műveletet.

- TWI Enable Acknowledge Bit (TWEA)

- Acknowledgment bit küldés engedélyezés. A TWI a megfelelő helyen ACK bitet generál, ha engedélyezve van. A letiltással ideiglenesen „le lehet csatlakoztatni” a buszról.



TWI regiszterek III.

- TWI Control Register (TWBR)

Bit	7	6	5	4	3	2	1	0	
	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	–	TWIE	TWCR
Read/Write	R/W	R/W	R/W	R/W	R	R/W	R	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- TWI START Condition Bit (TWSTA)

- Masterként lehet vele START conditiont generálni. Csak akkor kezd, ha a busz szabad. Szoftveresen törölni kell utána.

- TWI STOP Condition Bit (TWSTO)

- Masterként lehet vele STOP conditiont generálni. Automatikusan törlődik a végén. Slave módban hibából lehet visszatérni. A bit beírásának hatására a kontroller elengedi az SCLA és SDA vonalakat (nagyimpedanciás állapotba kerülnek).



TWI regiszterek IV.

- TWI Control Register (TWBR)

Bit	7	6	5	4	3	2	1	0	
	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	–	TWIE	TWCR
Read/Write	R/W	R/W	R/W	R/W	R	R/W	R	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- TWI Write Collision Flag (TWWC)
 - Beáll 1-be, ha a TWDR-t magas TWINT esetén próbáljuk írni.
- TWI Enable Bit (TWEN)
 - TWI engedélyezése. A TWI funkció felülvezérli a portlábakat.
- TWI Interrupt Enable (TWIE)
 - Ha 1, akkor a TWI megszakítás engedélyezve van.



TWI regiszterek V.

- TWI Status Register (TWSR)

Bit	7	6	5	4	3	2	1	0	
	TWS7	TWS6	TWS5	TWS4	TWS3	–	TWPS1	TWPS0	TWSR
Read/Write	R	R	R	R	R	R	R/W	R/W	
Initial Value	1	1	1	1	1	0	0	0	

- TWI Status (TWS7:3)

- A különböző módokban különböző státusz információkat ad. Részletek a kontroller leírásában a 212. oldaltól kezdve.

- TWI Prescaler Bits (TWPS1:0)

- Órajel előosztás
00 = 1, 01 = 4, 10 = 16, 11 = 64



TWI regiszterek VI.

- TWI Data Register (TWDR)

Bit	7	6	5	4	3	2	1	0									
	<table border="1"><tr><td>TWD7</td><td>TWD6</td><td>TWD5</td><td>TWD4</td><td>TWD3</td><td>TWD2</td><td>TWD1</td><td>TWD0</td></tr></table>								TWD7	TWD6	TWD5	TWD4	TWD3	TWD2	TWD1	TWD0	TWDR
TWD7	TWD6	TWD5	TWD4	TWD3	TWD2	TWD1	TWD0										
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W									
Initial Value	1	1	1	1	1	1	1	1									

- 8-bites adatregiszter küldésre és fogadásra. Értéke nem változik, amíg a TWINT értéke 1. Mindig a buszon lévő utolsó bájtot tartalmazza. Az ACK bitet a kontroller automatikusan kezeli, a CPU nem fér hozzá.



TWI regiszterek VII.

- TWI (Slave) Address Register (TWAR)

Bit	7	6	5	4	3	2	1	0	
	TWA6	TWA5	TWA4	TWA3	TWA2	TWA1	TWA0	TWGCE	TWAR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	1	1	1	1	1	1	1	0	

- TWI (Slave) Address (TWA)

- 7-bites eszközcím. Slave módban az erre a címre érkező adatokat fogja fogadni. Master módban nem kell használni.

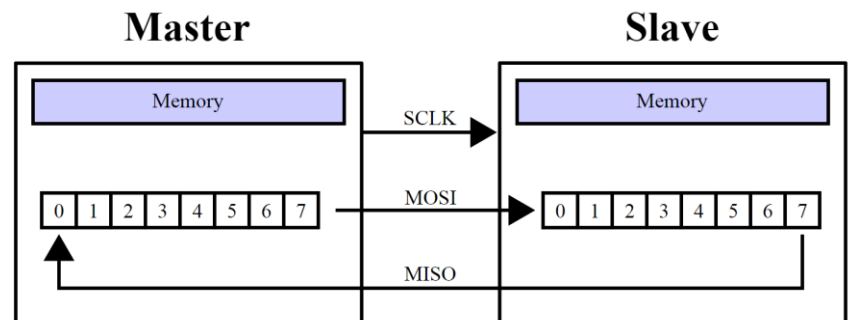
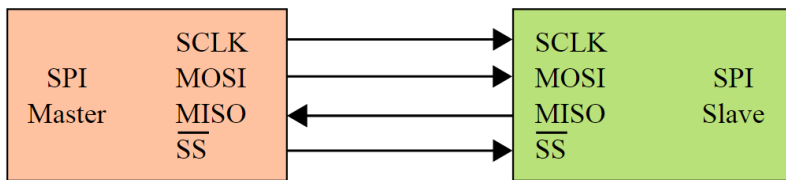
- TWI General Call Recognition Enable Bit (TWGCE)

- Ha 1, akkor engedélyezi a „General Call” felismerését.



SPI busz

- Az Serial Peripheral Interface (SPI) buszt a Motorola fejlesztett ki.
- Nagysebességű perifériák és a mikrovezérlők közötti kommunikációra használható, rövid távolságon (nyomtatott áramkörön határain) belül (tipikus példa az SD memóriakártya)
- Single-master, multi-slave, soros, szinkron buszrendszer
- 10-100 MHz órajel, eszköztől függően.
- Két shift regiszter van egy gyűrűben összekötve. Minden egyes kiküldött bithez tartozik egy fogadott bit.

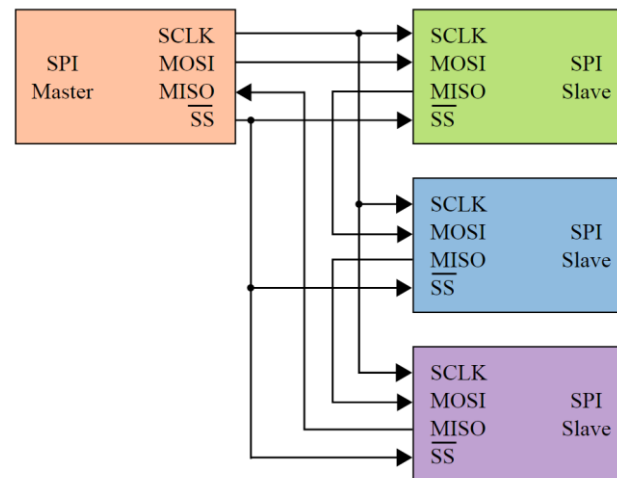
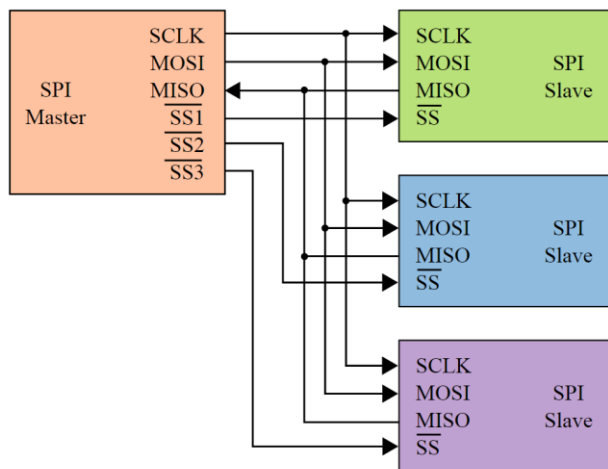


Forrás: [Wikipedia](#), [en:user:Cburnett](#), [CC BY-SA 3.0](#)



SPI hardver és topológia

- Minimálisan 4-vezetékes kialakítás
- „Push-pull” kimenet, így nem kell felhúzó ellenállás
 - SCLK: Serial Clock – közös órajelvezeték, a master generálja
 - MOSI: Master Output, Slave Input – közös master kimenet
 - MISO: Master Input, Slave Output – közös master bemenet
 - SS: Slave Select (active low) – slave engedélyező, master kimenet,
 - Függtelen slave-ek esetén minden slave-hez egy vezeték tartozik
 - Lánc (daisy chain) topológia esetén elég egy SS vezeték

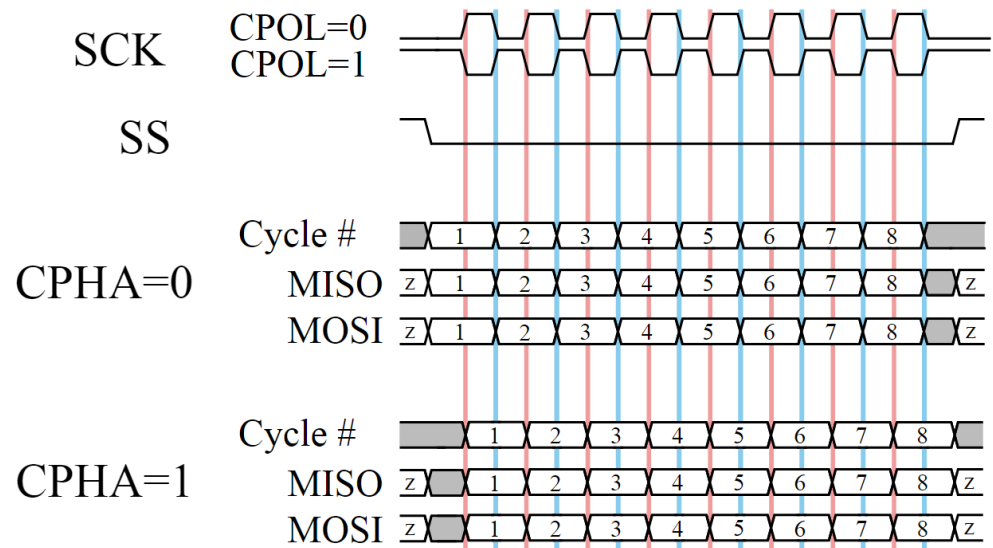


Forrás: [Wikipedia](#), [en:user:Cburnett](#), [CC BY-SA 3.0](#)



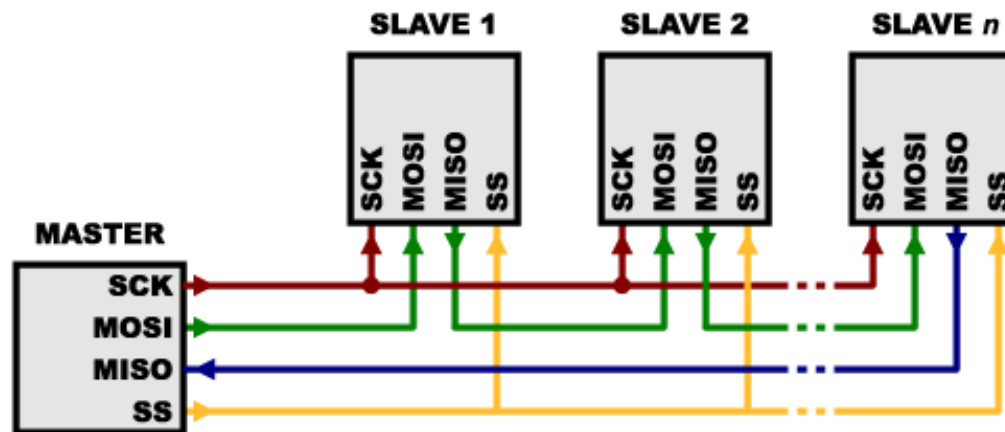
SPI adatátvitel (független slave-ek) I.

- Az órajel ütemezésre a MOSI lábon kiküldésre kerül a master shift regiszterének (jellemzően 8 vagy 16 bites) tartalma, és beolvasásra kerül az SS lábbal engedélyezett slave shift regisztere.
- A master által beállított órajel többféle lehet.
 - Az órajel polaritása (CPOL) kétféle
 - A fázis (CPHA) dönti el, hogy az első vagy a hátsó élre mintavételez-e
 - A beállításra figyelni kell!



SPI adatátvitel (daisy chain) II.

- Lánc topológia esetén a master kimenete (MOSI) az első slave bemenetéhez (MOSI) kapcsolódik.
- A slave-ek sorba vannak kötve, azaz az első slave kimenete a második slave bemenetéhez kapcsolódik, és így tovább.
- Végül az utolsó slave kimenete (MISO) a master bemenetéhez (MISO) kapcsolódik.
- Ilyenkor a slave-ek úgy működnek, hogy a fogadott adatot továbbadják (shiftelik) a következő slave-nek és végül a masternek.
- Az egész rendszer egy kommunikációs shift regiszterkén működik.



Forrás: [SparkFun Electronics](#), [MIKEGRUSIN](#), [CC BY-NC-SA 3.0](#)



SPI regiszterek I.

- SPI Control Register (SPCR)

Bit	7	6	5	4	3	2	1	0									
	<table border="1"><tr><td>SPIE</td><td>SPE</td><td>DORD</td><td>MSTR</td><td>CPOL</td><td>CPHA</td><td>SPR1</td><td>SPR0</td></tr></table>								SPIE	SPE	DORD	MSTR	CPOL	CPHA	SPR1	SPR0	SPCR
SPIE	SPE	DORD	MSTR	CPOL	CPHA	SPR1	SPR0										
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W									
Initial Value	0	0	0	0	0	0	0	0									

- SPI Interrupt Enable (SPIE)
 - Ha 1, akkor az SPI megszakítás engedélyezve van.
- SPI Enable (SPE)
 - SPI engedélyező bit.
- Data Order (DORD)
 - Ha 1, akkor az LSB-tel kezdi a küldést, 0 esetén a MSB-tel.
- Master/Slave Select (MSTR)
 - Ha 1, akkor master módban van a kontroller, ha 0 akkor slave módban.



SPI regiszterek II.

- SPI Control Register (SPCR)

Bit	7	6	5	4	3	2	1	0									
	<table border="1"><tr><td>SPIE</td><td>SPE</td><td>DORD</td><td>MSTR</td><td>CPOL</td><td>CPHA</td><td>SPR1</td><td>SPR0</td></tr></table>								SPIE	SPE	DORD	MSTR	CPOL	CPHA	SPR1	SPR0	SPCR
SPIE	SPE	DORD	MSTR	CPOL	CPHA	SPR1	SPR0										
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W									
Initial Value	0	0	0	0	0	0	0	0									

- Clock Polarity (CPOL)

- Órajel polaritás. Lásd korábban!

- Clock Phase (CPHA)

- Órajel fázis. Lásd korábban!

- SPI Clock Rate Select (SPR1:0)

- 00 = $f/4$, 01 = $f/16$, 10 = $f/64$, 11 = $f/128$



SPI regiszterek III.

- SPI Status Register (SPSR)

Bit	7	6	5	4	3	2	1	0	
	SPIF	WCOL	–	–	–	–	–	SPI2X	SPSR
Read/Write	R	R	R	R	R	R	R	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- SPI Interrupt Flag (SPIF)

- SPI megszakítás jelzőbit. Hardveresen törlődik a megszakítás kezelésekor. Master módban akkor is beáll, ha az SS lábat alacsonyra húzták.

- Write Collision Flag (WCOL)

- Ütközést jelző bit. Beáll, ha az SPDR regiszert írják küldés közben.

- Double SPI Speed Bit (SPI2X)

- Megduplázza az SCK frekvenciát, azaz megfelel az előosztást.



SPI regiszterek IV.

- SPI Data Register (SPDR)

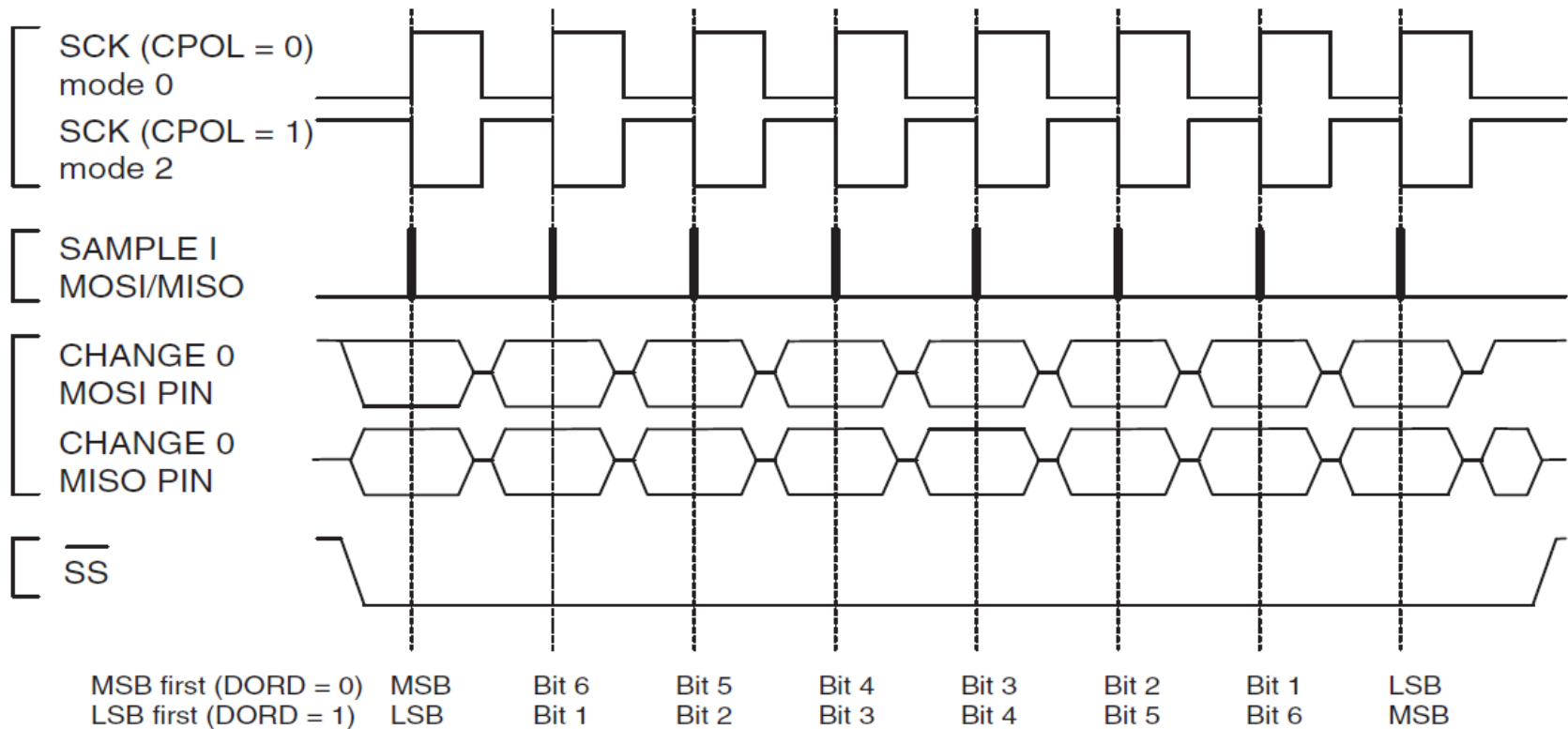
Bit	7	6	5	4	3	2	1	0	
	MSB							LSB	SPDR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	X	X	X	X	X	X	X	X	Undefined

– Küldő és fogadó regiszter. Reset után nem definiált.



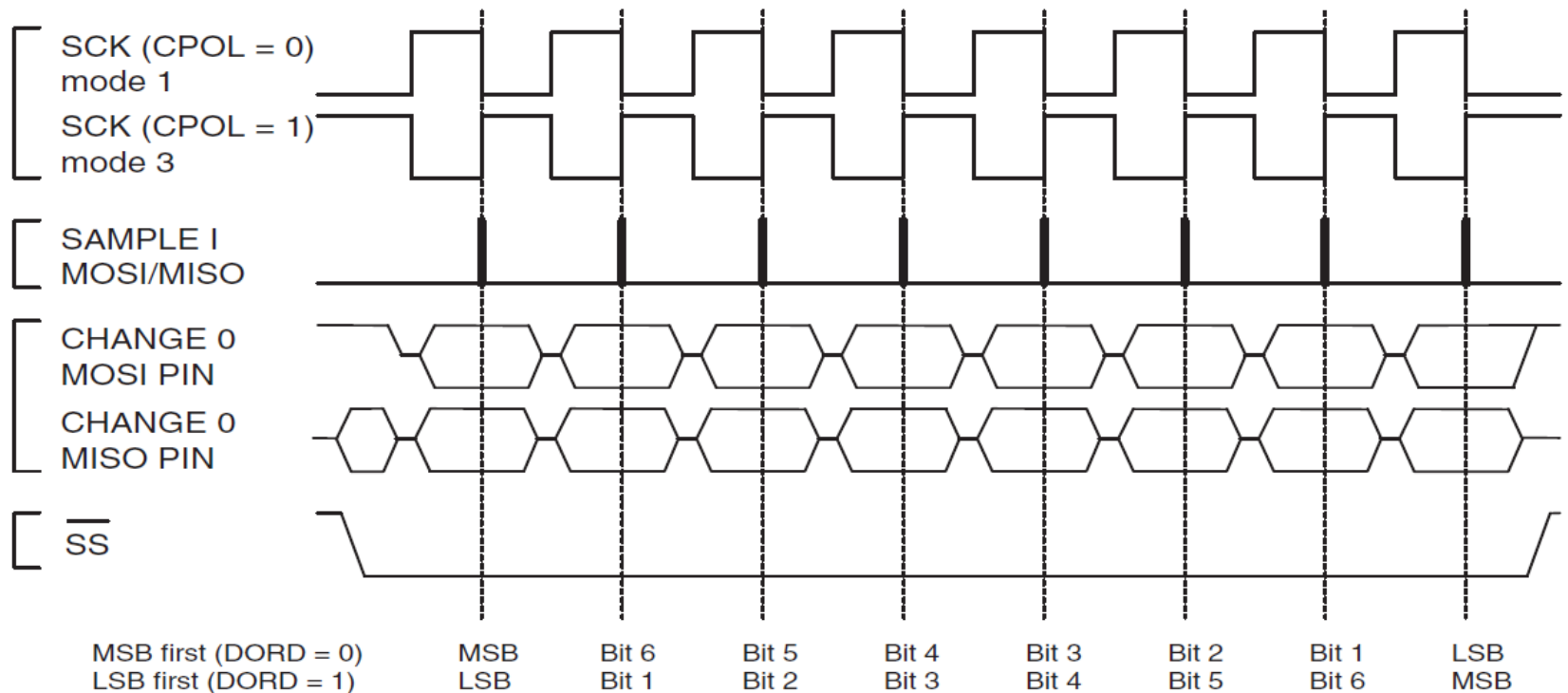
SPI jelalakov I.

- CPHA = 0



SPI jelalakov II.

- CPHA = 1



BUDAPESTI MŰSZAKI ÉS GAZDASÁGTUDOMÁNYI EGYETEM

Dr. Aradi Szilárd



email: aradi.szilard@mail.bme.hu

Köszönöm a figyelmet!



BME KÖZLEKEDÉSMÉRNÖKI ÉS JÁRMŰMÉRNÖKI KAR
32708-2/2017/INTFIN SZÁMÚ EMMI ÁLTAL TÁMOGATOTT TANANYAG