

A mikroszámítógép és a PC-s applikáció közti UART interfész

A mikroszámítógép és egy PC-n működő applikáció között soros vonali (UART) interfészen keresztül történik kommunikáció. Az UART interfész paraméterei:

- Baud Rate: 921 600 bit / s
- Adathossz: 8 bit
- Stop bitek száma: 1
- Paritás: nincs

A kommunikáció ASCII formában valósul meg, azaz mindkét irányban egy-egy karakter sorozat CR karakterrel lezárva (egy sor) képezi a kommunikáció egységét. A karakter-sorozatot egy '\$' kezdő karakter indítja. Egy üzenet-sor formátuma a következő:

\$<code><arg1><arg2>...<argn>CR

- A <code> paraméter 1-5 karakter hosszúságú üzenet-kód, amely meghatározza az üzenet további szerkezetét is.
- Az üzenet-kódot maximálisan 5 argumentum követi (beleértve a 0 számú argumentumot is), amelyek az üzenet-típusban rögzített formátumúak és hosszúságúak. Köztük nincs elválasztó karakter.
- Ha az üzenetnek egy argumentuma van, az lehet tetszőleges hosszúságú decimális egész vagy tört szám, továbbá karakter sorozat (string):
 - decimális egész szám esetén általában egy karakteres, azaz 0 és 9 közé eső szám az argumentum,
 - tizedes tört esetén előjeles egész részt, tizedespontot és törtrészt tartalmazó szám,
 - string esetében ez lehet maximálisan 80 karakter hosszúságú üzenet sor.
- Általános esetben egy argumentum formátuma: 2, 4 vagy 8 karaktert tartalmazó hexadecimális szám szükség szerint '0'-val bevezetve. Ez maga után vonja, hogy az illető számot előjel nélküli egész számként kell továbbítani. A különböző formátumú (a jelen feladat szempontjából releváns) számokat a következő kódolással lehet ilyen formára hozni:
 - előjeles 16-bites egész: `*(uint16_t *)&nbr,`
 - előjeles 32-bites egész: `*(uint32_t *)&nbr,`
 - 32-bites lebegőpontos (float) adat: `*(uint32_t *)&nbr.`

A mikroszámítógépen a soros vonalra küldés a 'sprintf' függvény alkalmazásával történik. A hexadecimális formátum előállítása a „%02X”, „%04X”, „%08X” formátum string-ekkel történik a 1, 2, ill. 8 byte terjedelmű adatokra nézve.

A következőkben a PC-ről a mikroszámítógép illetve a mikroszámítógépről a PC felé irányuló üzeneteket specifikáljuk.

PC → μC

Command				Function	#	Argument(s)	Format
D	P	1		Potentiometer 1 Measurement Value Request			
		2		Potentiometer 2 Measurement Value Request			
	S			Switches State Request			
	L			LEDs State Request			
	B			Battery Voltage Request			
R	P			Send Remote Potentiometer Value	1	int16_t remPotVal	%04X
C	C	T		Select Control Type	1	int motorCtrlType	%d
		P		Set Control Sample Period	1	uint16_t samplePeriod	%04X
	R	P	P	Set Motor RPM Control Proportional Gain	1	%f<double mcRMPPropGain	%04X
			C	Set RPM Control Proportional Gain Correction On/Off	1	uint8_t mcRMPPropGainCorr	%02X
			G	Set Motor RPM Control PI Gain	f	float mcRMPPIGain	%f
			I	Set Motor RPM Control PI Integration Time	1	float mcRMPPIti	%f
	S	P	P	Set Motor Servo Control Proportional Gain	1	float mcServoPropGain	%f
			E	Set Motor Servo Control PIRPM On/Off	1	int mcServoEmbPIRPM	%d
			G	Set Motor Servo Control PIRPM Gain	1	float mcServoPIRPMGain	%f
			I	Set Motor Servo Control PIRPM Integration Time	1	float mcServoPIRPMti	%f
	L	A	0	Set General Linear Control Ac(0,0)	1	float mcglAc[0][0]	%f
				Set General Linear Control Ac(0,1)	1	float mcglAc[0][1]	%f
				Set General Linear Control Ac(0,2)	1	float mcglAc[0][2]	%f
			1	Set General Linear Control Ac(1,0)	1	float mcglAc[1][0]	%f
				Set General Linear Control Ac(1,1)	1	float mcglAc[1][1]	%f
				Set General Linear Control Ac(1,2)	1	float mcglAc[1][2]	%f
		2	0	Set General Linear Control Ac(2,0)	1	float mcglAc[2][0]	%f
				Set General Linear Control Ac(2,1)	1	float mcglAc[2][1]	%f
				Set General Linear Control Ac(2,2)	1	float mcglAc[2][2]	%f
		B	0	Set General Linear Control Bc(0)	1	float mcglBc[0]	%f
			1	Set General Linear Control Bc(1)	1	float mcglBc[1]	%f
			2	Set General Linear Control Bc(2)	1	float mcglBc[2]	%f
		C	0	Set General Linear Control Cc(0)	1	float mcglCc[0]	%f
			1	Set General Linear Control Cc(1)	1	float mcglCc[1]	%f
			2	Set General Linear Control Cc(2)	1	float mcglCc[2]	%f
		D		Set General Linear Control Dc	1	float mcglDc	%f
T	R			Reset Setpoint Signal Storage			
	L			Load Setpoint Signal Data	1	int16_t sigData	%04X
	E			End Load Test Signal Data	1	uint16_t testSigSize	%04X

μC → PC

Command		Function	#	Argument(s)	Format
D	P	1	Send Potentiometer 1 Measurement Value	1 int16_t pot1Val	%04X
	P	2	Send Potentiometer 2 Measurement Value	1 int16_t pot2Val	%04X
	S		Send Switches State	1 uint16_t switchState	%04X
	L		Send LEDs State	1 uint16_t ledsState	%04X
	M		Send Message Line	1 char message[]	%s
	E		Send Error	2 uint16_t errorID uint32_t errInfo	%04X%08IX
I			Initialization Request		
C	D		Send Motor Control Data Values	5 int16_t dcmCtrlSetp int16_t motorSpeed int16_t motorAngle int16_t motorActVal uint16_t remTstamp	%04X%04X%04X%04X%04X
S	A		Send Angle Sensor Error State	1 uint8_t asANGsensErr	%02X
B	V		Send Battery Voltage	1 uint16_t vbat_volt * 100 (vbat_volt = (float)vbatVal / 4096.0 * 3.3 * 9.3 / 1.8)	%04X
T	S		Test Start		
	E		Test End		
	R		Test Reset		

Összehangolt inicializálás a mikroszámítógépes és PC program között

Lényeges, hogy akár a mikroszámítógépes, akár a PC program indításakor összehangolt állapot álljon elő a két rendszerben.

A PC applikáció mindenképpen magasabb prioritást képvisel, tehát akármelyiket is indítjuk előbb, a mikrogépes rendszernek kell alapvetően felvennie a PC applikáció állapotát. A PC applikáció megőrzi a korábbi futtatás paramétereit, továbbá lehetőség van ezek átírására a mikrogépes program indítása előtt. Ugyanakkor a PC applikációnak értesülnie kell a mikrogépes rendszer csak ott beállítható vagy befolyásolható egységeinek állapotáról. Ennek fő összetevői a mikrogépes rendszer két potenciométerének állása, kapcsolóinak, és LED alapú megjelenítőinek állapota, továbbá a bejövő motor-tápfeszültség mért értéke.

Ha mikroszámítógépes programot a futó PC alkalmazás mellett indítjuk, akkor az elküld egy '\$I' (Initialization Request) üzenetet a PC-nek. Ennek hatására a PS-s applikáció egy inicializáló üzenetsort küld el a mikrogépnek.

Ha működő mikrogépes program mellett indítjuk el a PC-s alkalmazást, akkor egy alkalommal lefut ugyanaz az inicializáló üzenetsorozat.

Ennek tartalma:

- \$DP1, \$DP2, \$DS, \$DL, \$BV üzenetek elküldése a mikrogépes rendszer kezelőszerveinek lekérdezéseire,
- \$TR – az mikrogépen működő bármilyen tesztfolyamat alaphelyzetbe állítására,
- \$RP – a PC-s applikációban realizált Remote Potentiometer értékének közlésére,
- \$CCT – az Irányítási Séma beállítására,
- \$CCP – a mintavételi periódusidő beállítására,
- \$CRPPG, \$CRPCE, \$CRPIG, \$CRPII, \$CSPPG, \$CSPIE, \$CSPIG, \$CSPII, \$CLA00, \$CLA01, \$CLA02, \$CLA10, \$CLA11, \$CLA12, \$CLA20, \$CLA21, \$CLA22, \$CLB0, \$CLB1, \$CLB2, \$CLC0, \$CLC1, \$CLC2, \$CLD üzenetek alkalmazásával az egye irányítási sémákhoz tartozó paraméterek beállítására.

Válaszul a mikrogépes program elküldi a potenciométerek, a kapcsolók és a LED-ek állapotát, valamint a kezdetben mért motorfeszültség értékét.

Lényeges lenne még a mért motor-fordulatszám és a tengely szöghelyzet mért értékének megadása, de ez az irányítás elindulásakor azonnal keletkezik és elküldésre kerül.