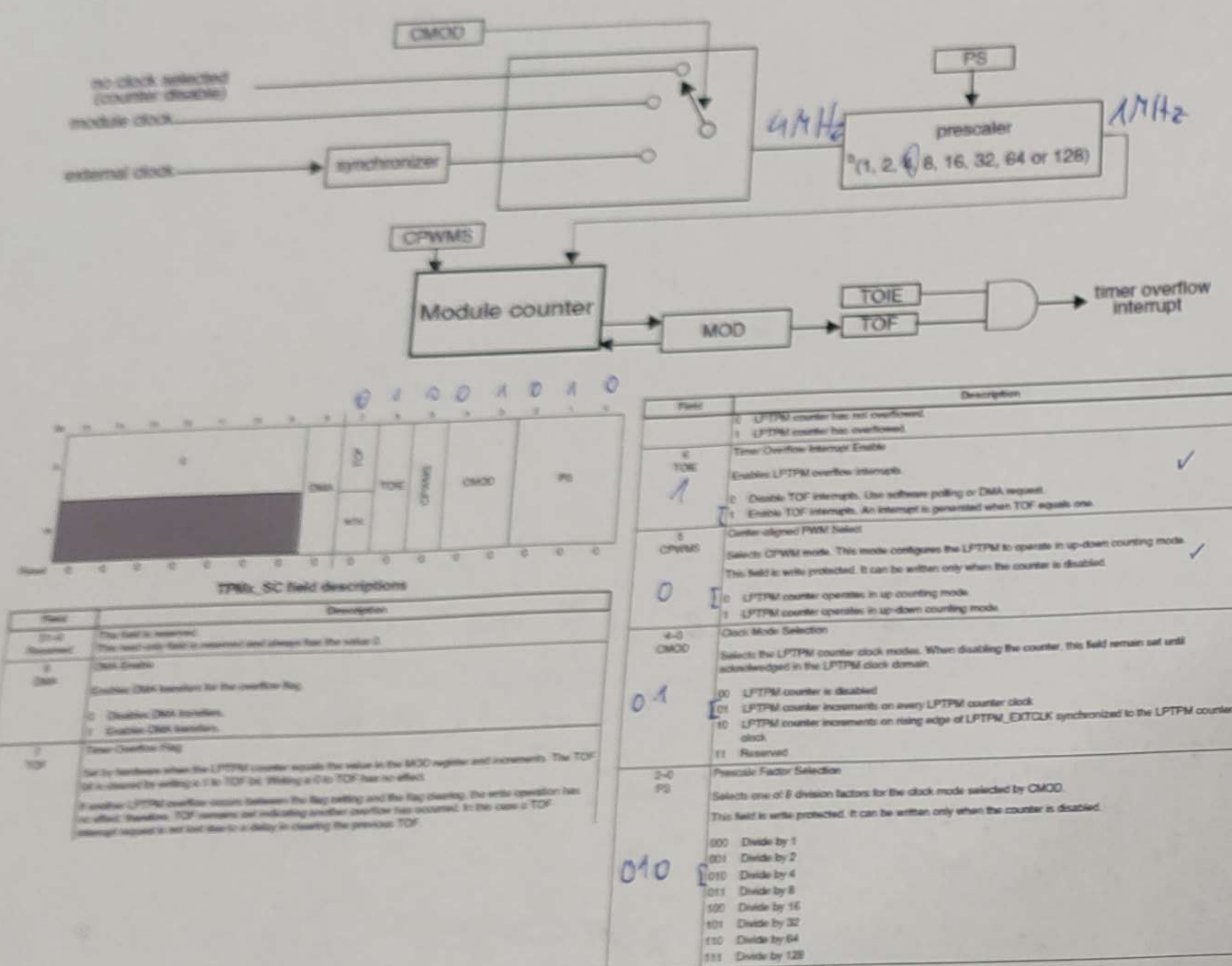


5. Časovač TPM0 má na vstupu hodinový signál 4MHz, v registru TPM0_SC nastavenou hodnotu 0x4A. Určete hodnotu registru TPM0_MOD (modulo hlavního čítače) tak, aby přerušení od přetečení čítače bylo generováno každých 6ms. Hodnotu vepište do připraveného rámečku hexadecimálně. Pamatuje, že čítač časovače je 16 bitový. Dále uveďte, jakým příkazem v C potvrdíte toto přerušení na začátku každé obsluhy přerušení.



1 000 000 taktů za sekundu, chceme teda int. každých 6 000. taktů → mod = 6000 - 1 = 5 999 = 0x176F

TPM0_MOD = 0x176F

Potvrzení přerušení od přetečení: ~~TPM0~~ TPM0 → SC |= (1 << 7);

C - všichni sme mali túto skupinu
Jméno: Patrik Skaloš
Podpis:

login: xskalo01

Pořadí: 346

Závěrečná zkouška z IMP 17. 1. 2023 16:00

Příklad	1	2	3	4	5
Body	10	12	10	9	10

Pokyny ke zkoušce:

Maximální čas na vypracování zkoušky je 90 minut. Skončíte-li dříve, tiše odevzdejte, nerušte ostatní.

Povoleny jsou psací potřeby, čistý papír na pomocné výpočty, kalkulačka.

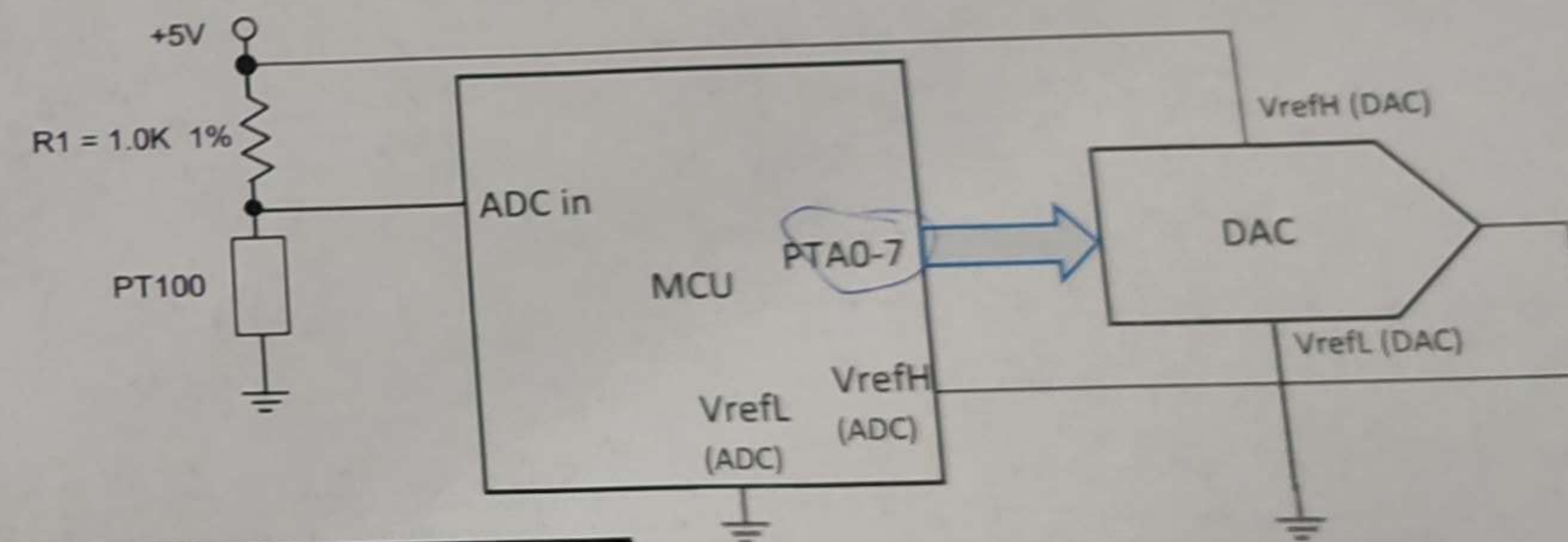
Zákázáno je používat cokoli mimo výše uvedené, zejména jakékoli komunikační prostředky, mobilní telefony, fotoaparáty nebo záznamová zařízení.

Každý pracuje samostatně.

Porušení výše uvedeného vede na hodnocení zkoušky 0 body a předání případu disciplinární komisi.

Výsledky pište nebo kreslete čitelně přímo do tohoto dvojlistu vždy v takovém formátu, jak je požadováno v příkladu. Lze vložit pomocný papír (označený jménem a loginem).

1. Referenční napětí pro ADC modul je generováno osmibitovým DA převodníkem (DAC), jehož referenční napětí je 5 V. Na vstup ADC je připojeno RTD čidlo teploty PT100. Zapojení je na obrázku. Čidlo PT 100 má odpor dle tabulky. Určete hodnotu vstupního čísla pro DA převodník (hodnotu, kterou zapíšete do datového registru portu PTA), určující referenční napětí ADC modulu tak, aby byl co nejlépe využit rozsah ADC na rozsahu teplot do 350°C. Hodnotu uveďte hexadecimálně.



Temperature Celsius	PT100 Resistance
-40	85
-20	92
0	100
20	108
100	138
200	176
300	214
350	233

$$V_{ADC} = V_{REF} \cdot \frac{R_2}{R_1 + R_2} = 5 \cdot \frac{233}{1233} = 0,94485 V$$

~~V_{ADC} = 0,94485 V~~

$$R_{odkazenie\ DAC} = \frac{5V}{2^8} = 0,019531 V / \text{bit}$$

$$V_{vstup\ DAC} = \lceil 0,94485V / 0,019531V \rceil = \lceil 48,37 \rceil = 49$$

$$(49)_{10} = 0x31$$

Hodnota na PTA0-7: 0x31

3. Určete, kolik termoelektrických generátorů TGP-751 bude třeba pro napájení senzorového uzlu bezdrátové sítě energy harvestingem za pomoci těchto generátorů, jestliže předpokládáme, že jeden generátor dokáže dodávat výkon asi 2 mW. Senzorový uzel sestává z mikrokontroléru s připojeným snímačem teploty a bezdrátovým modulem. Mikrokontrolér se každou sekundu probudí z režimu VLLS1 do běhu včetně periférií, provede měření a přepočítá, což potřebuje 15000 taktů hodin jádra a poté odvyšlá naměřené teploty. Pak se zase uspi do režimu VLLS1. Vysílání trvá 1 ms a odebírá se při tom 35 mA. Napájecí napětí systému je 3 V, systém je na studené straně generátoru, kde se předpokládá teplota 25 °C.

Symbol	Description	Temp.	Typ.	Max	Unit
I _{DDA}	Analog supply current	—	—	See note	mA
I _{DD RUN}	Run mode current - 48 MHz core / 24 MHz bus and flash, all peripheral clocks enabled, code executing from flash, at 3.0 V	at 25 °C at 95 °C	4.8 5	5 5.2	mA
I _{DD VLLS1}	Very low-leakage stop mode 1 current at 3.0 V	at 25 °C at 50 °C at 70 °C at 85 °C	0.58 0.9 1.68 3.51	0.69 1.04 2.02 4.05	µA

Symbol	Description	Min.	Typ.	Max.	Unit
	VLLS1 → RUN	—	93	115	µs

Probudenie = 115 µs @ 5 mA = 0,000 575 mAs

Meranie = 15k taktov @ 5 mA = 312,5 µs @ 5 mA = 0,001 562 5 mAs

Vysielanie = 1 ms @ 35 mA = 0,035 mAs

Spolu = 0,037 137 5 mAs

Spolu = 0,037 137 5 mAs + 0,001 562 5 mAs + 0,035 mAs = 0,073 700 mAs

Spolu spotreba = 0,073 700 mAs @ 3 V = 0,221 1 mW

Spolu RUN = prebudenie + meranie = 427,5 µs @ 5 mA = 0,002 137 5 mAs

Spolu sleep = 999,572 5 ms na 0,69 µA = 0,000 689 705 mAs

Dokopy za sekundu = (RUN + sleep + vysielanie) @ 3 V = 0,221 1 mW

Energie potrebná pro probuzení [mAs]: 0,000 575

Energie potrebná pro běh [mAs]: 0,001 562 5 = meranie, bez prebudenia a periférii

Energie potrebná pro odeslání [mAs]: 0,035

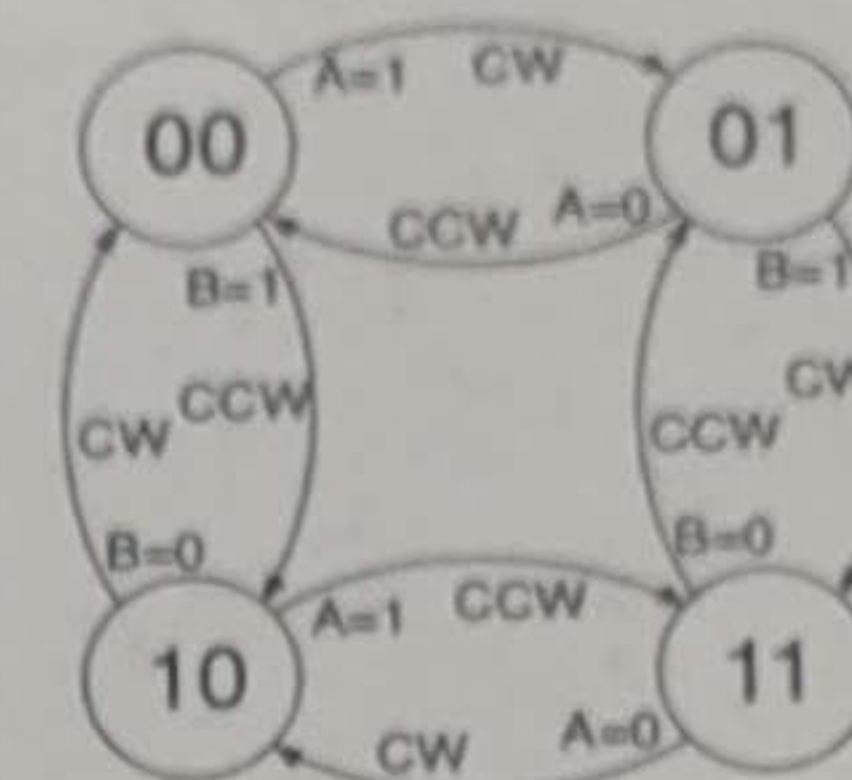
Energie spotřebovaná ve VLLS1 za sekundu [mAs]: 0,000 689 705

Energie celkem za 1 s [mWs]: 0,113 481

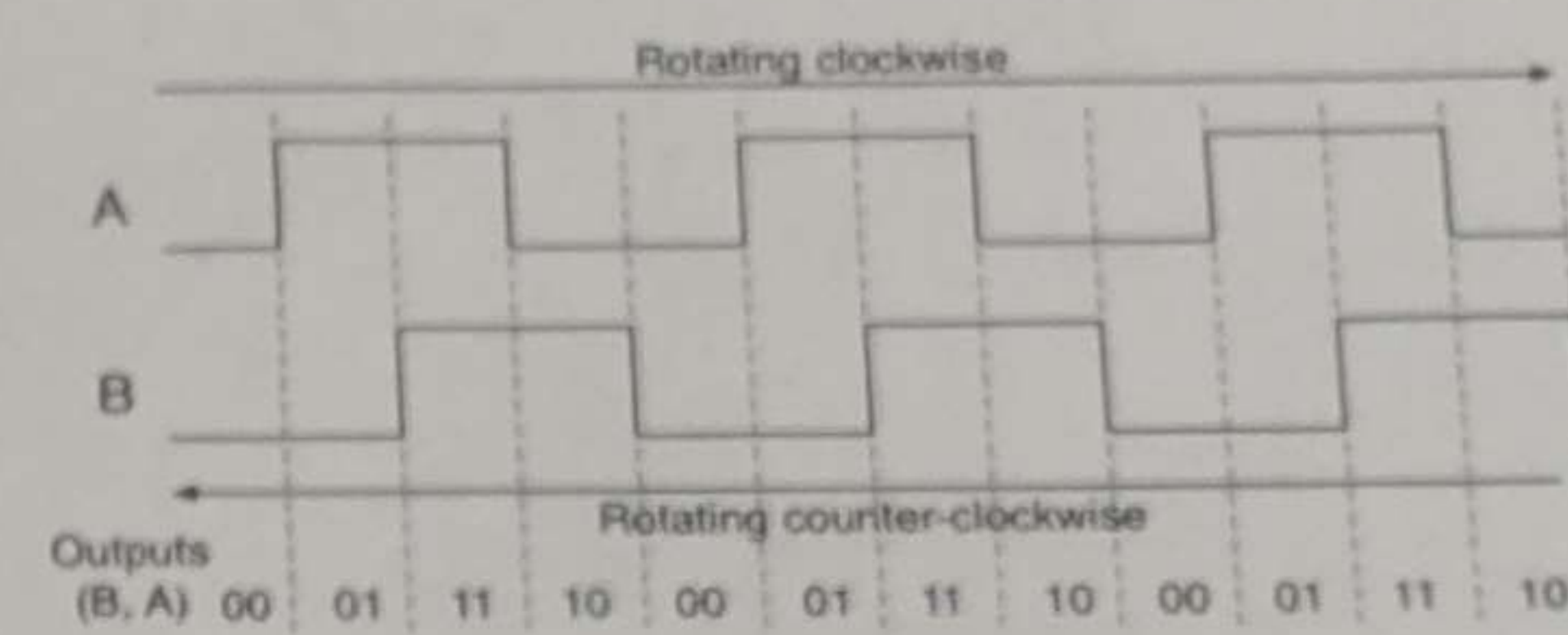
Kolik generátorů TGP-751 bude třeba: 205 (?)

za hodinu spotřebuje 408,5316 mWh? nevím počítat s As a Ws :-)

3. Napište v C kód přerušovací rutiny od GPIO pinů portu A, kde na piny PTA0 a PTA1 jsou připojeny dva kanály (fáze A a B) inkrementálního snímače polohy. Přerušení je nastaveno tak, že je vyvoláno při libovolné změně stavu na pinech. Absolutní poloha je v globální proměnné position, kterou krok po směru hodinových ručiček (clockwise, CW) má inkrementovat (+1), krok proti směru (counterclockwise, CCW) dekrementovat (-1). Mezní stavy proměnné nemusíte ošetřovat. Pro získání stavu pinů použijte masku na registr GPIOA_PDIR. Řešení stačí naznačit pro jeden či dva stavy.



State diagram for a rotary encoder

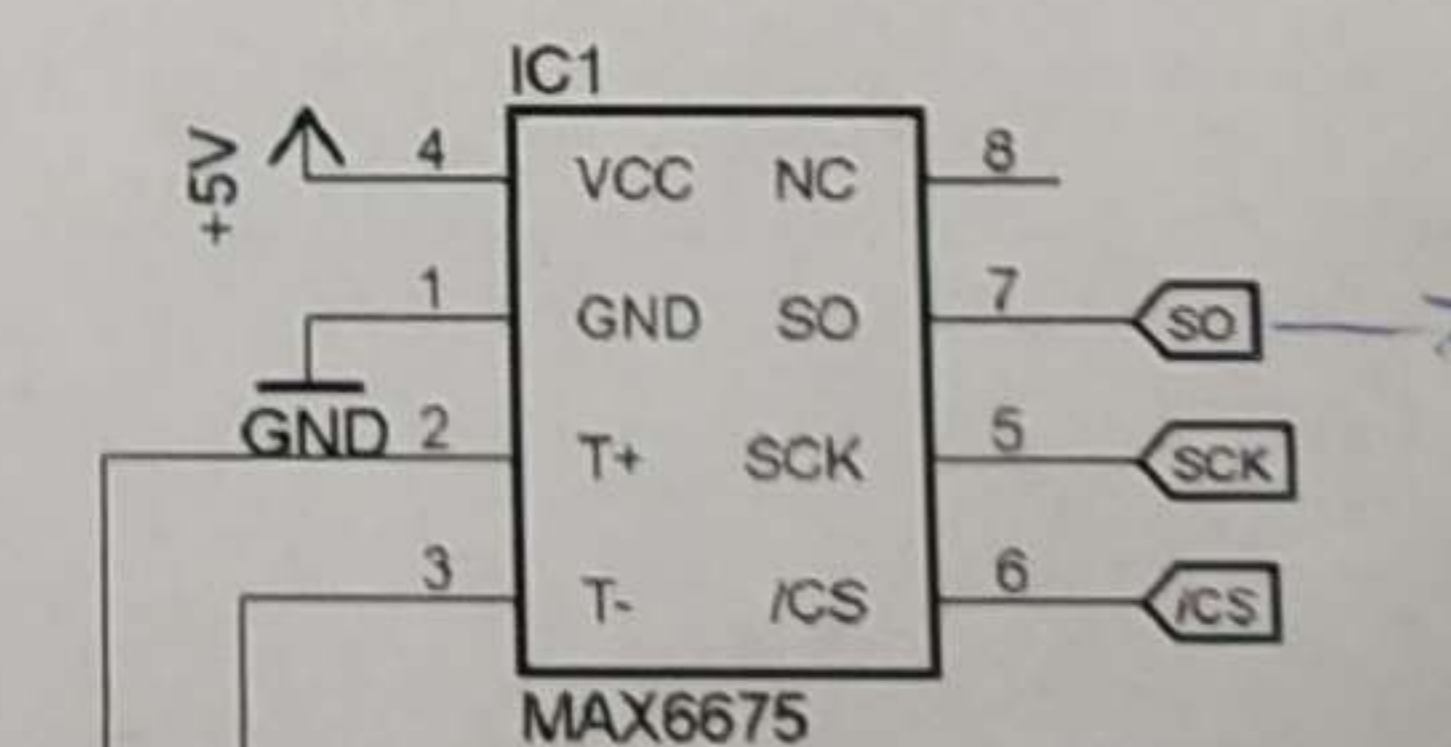


Output of the pins of an encoder while being rotated

int position=0;

```
void PORTA_IRQHandler(void) {
    int A = GPIOA->PDIR & 1; // bit 0
    int B = GPIOA->PDIR & 2; // bit 1
    if (GPIOA->IR & 1) { // Port 0 caused interrupt, IR = interrupt reg.
        GPIOA->CIR = 1; // Clear interrupt register - clear INTR for port 0
        A_prev = A > 0 ? 0 : 1; // Previous value was opposite
    } else if (GPIOA->IR & 2) {
        GPIOA->CIR = 2;
        B_prev = B > 0 ? 0 : 1;
    }
    if (A_prev == 0 && A != 0 && B == 0 || A_prev == 1 && A != 1 && B == 1 || B_prev == 0 && B != 0 && A == 0 || B_prev == 1 && B != 1 && A == 1) {
        position++;
    } else {
        position--;
    }
}
```

4. Na rozhraní SPI0 je připojen obvod rozhraní termočlánu MAX6675 jako slave, který pouze vysílá naměřená data do mikrokontroléru (ten je master). Termočlánek je připojen na piny T+ a T-, rozhraní SPI tvoří SO (data, do MCU připojeno na pin MISO), SCK (hodiny, které generuje master, tedy MCU) a /CS (do MCU jako Slave Select, SS) Uvedte, jak v programu (jakým příkazem v C například) zahájíte přenos dat ze snímače do mikrokontroléru. Datový registr SPI0 se jmenuje SPI0_D.



Malo by stačit na pin SS strany MCU (master) zapísat logickou nulu. Z daného datového registru potom MCU iba číta, nie zapisuje.

(zapísať logickú nulu, lebo slave select nesie byť log. 0, ale so slave-om nekomunikujeme. Privedenie log. 0 by mal zariadeniu naznačiť, že môže vyslať data)