

Technika cyfrowa i mikroprocesorowa

Tryby oszczędzania energii w mikrokontrolerach

Wojciech Tarnawski

www.w-tarnawski.pl

wojciech.tarnawski@pwr.edu.pl



HR EXCELLENCE IN RESEARCH



Politechnika Wrocławska

Tryby oszczędzania energii

- Co to jest oszczędzanie energii?
- Czy zawsze jest to potrzebne?
- Prosty sposób oszczędzania energii
- Tryby oszczędzania energii Atmega328P
- Atmega328P vs Arduino UNO
- Tryby oszczędzania energii – jak to stosować?
- Obliczanie czasu pracy na baterii
- Zabezpieczenie BOD – brown-out detection
- Raspberry PI

Tryby oszczędzania energii

Co to jest oszczędzanie energii?

Próba optymalizacji działania układu/programu pod względem efektywności energetycznej, bez utraty funkcjonalności.

Staramy się używać trybów oszczędzania energii dostępnych w wybranym mikrokontrolerze. Projektując elektronikę wybieramy układy energooszczędne.

Czasem stworzenie programu oszczędzającego energię wymaga innego podejścia przy tworzeniu kodu. Może zdarzyć się tak, że w skrajnych przypadkach do przepisania będzie cały program. Dlatego należy już na samym początku projektu ustalić jakie są warunki zasilania i czego klient oczekuje.

Jeśli założenia czasu działania i zasilania nie są możliwe do spełnienia to należy zastanowić się nad sensem projektu. Dokonać modyfikacji założeń.

Tryby oszczędzania energii

Czy zawsze jest to potrzebne?

Bateria 100mAh

Zużycie [mA]	Czas pracy [h]
1000	0,1
500	0,2
250	0,4
100	1
50	2
10	10
5	20
1	100
0,5	200

Bateria 1000mAh

Zużycie [mA]	Czas pracy [h]
1000	1
500	2
250	4
100	10
50	20
10	100
5	200
1	1000
0,5	2000

Bateria 10000mAh

Zużycie [mA]	Czas pracy [h]
1000	10
500	20
250	40
100	100
50	200
10	1000
5	2000
1	10000
0,5	20000

Tryby oszczędzania energii

Czy zawsze jest to potrzebne?

NIE

ale zawsze warto

Tryby oszczędzania energii

Czy zawsze jest to potrzebne?

NIE

- nie jest to główny problem jak mamy zasilanie zewnętrzne
- aplikacja ma pracować bez przerw, sterowanie procesami ciągłymi, systemy czasu rzeczywistego itd.
- jeśli pobór energii przez układ sterowania jest dużo mniejszy niż energia potrzebna na pozostałe elementy (np. drony latające – silniki pobierają znaczącą część energii)
- ograniczony budżet – trudniejszy program wymaga więcej nakładu pracy

Tryby oszczędzania energii

Czy zawsze jest to potrzebne?

Warto

- energia zawsze kosztuje - oszczędności finansowe
- mniej energii używanej – mniejsze nagrzewanie układu – łatwiejsze chłodzenie
- aspekty ekologiczne – ciepło, energia
- dodatkowa praca dla programisty - wynagrodzenie

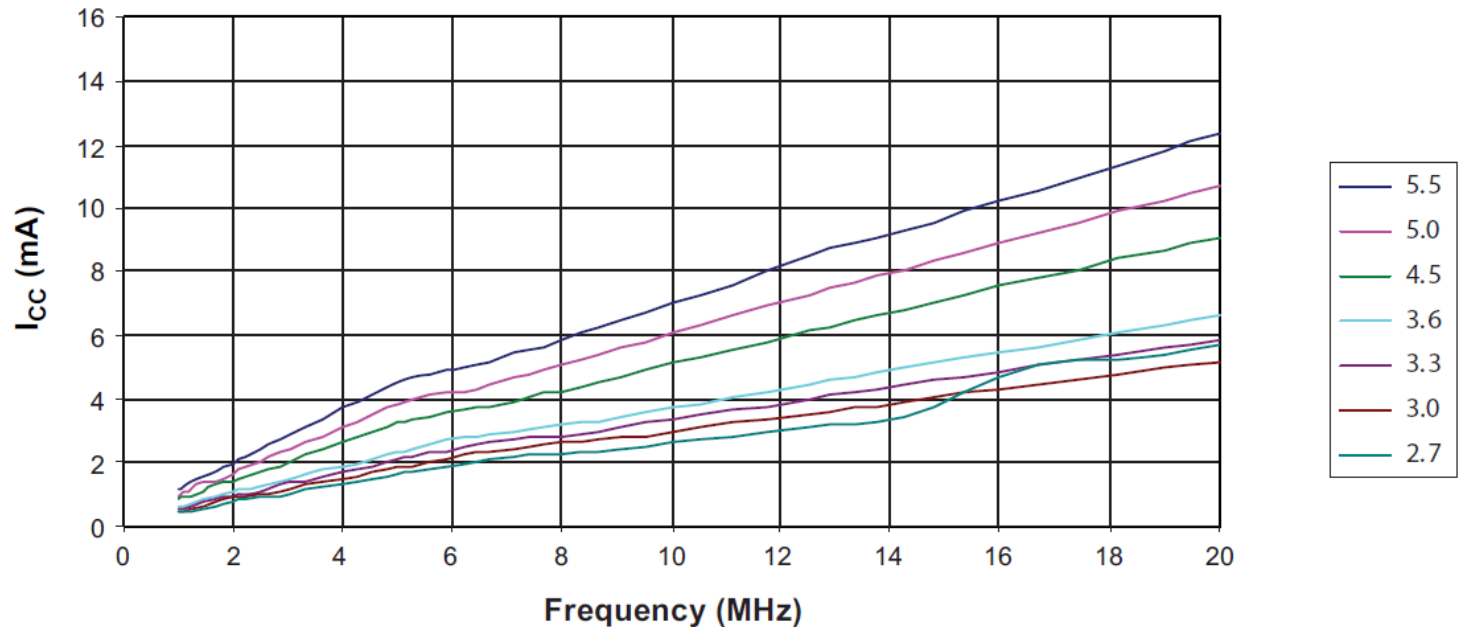
Pobór prądu [A]	0,1	Pobór prądu [A]	0,1	Pobór prądu [A]	1
Napięcie układu [V]	5	Napięcie układu [V]	5	Napięcie układu [V]	5
Moc układu [W]	0,5	Moc układu [W]	0,5	Moc układu [W]	5
Ilość urządzeń	1	Ilość urządzeń	10	Ilość urządzeń	10
ilość godzin w miesiącu [30dni]	720	ilość godzin w miesiącu [30dni]	720	ilość godzin w miesiącu [30dni]	720
Zużycie energii [Wh]	360	Zużycie energii [Wh]	3600	Zużycie energii [Wh]	36000
Cena [kWh-zł]	1,00 zł	Cena [kWh-zł]	1,00 zł	Cena [kWh-zł]	1,00 zł
Koszt miesięczny	0,36 zł	Koszt miesięczny	3,60 zł	Koszt miesięczny	36,00 zł
Koszt roczny	4,32 zł	Koszt roczny	43,20 zł	Koszt roczny	432,00 zł

Tryby oszczędzania energii

Prosty sposób oszczędzania energii

Atmega328P

Figure 29-1. Active Supply Current versus Frequency



Tryby oszczędzania energii

Atmega328P

- Special microcontroller features
 - Power-on reset and programmable brown-out detection
 - Internal calibrated oscillator
 - External and internal interrupt sources
 - Six sleep modes: Idle, ADC noise reduction, power-save, power-down, standby, and extended standby

- Low power consumption
 - Active mode: 1.5mA at 3V - 4MHz
 - Power-down mode: 1 μ A at 3V

Tryby oszczędzania energii

Atmega328P

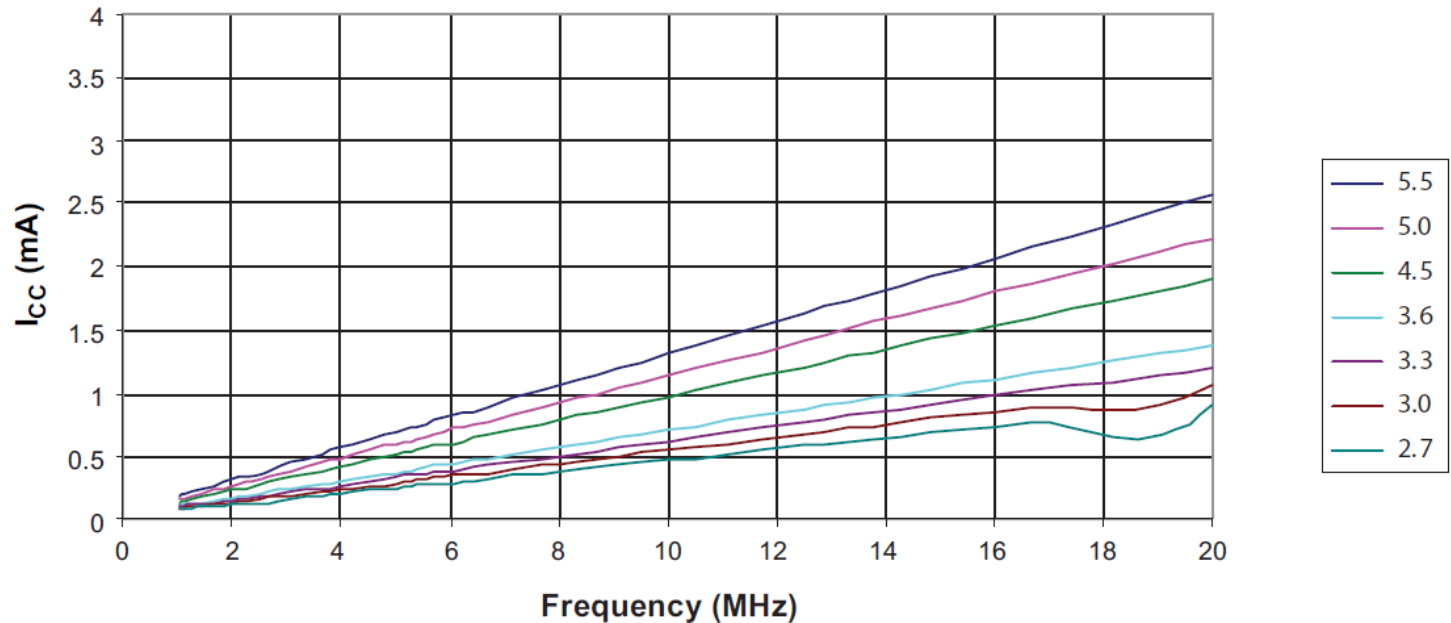
Table 9-1. Active Clock Domains and Wake-up Sources in the Different Sleep Modes.

Sleep Mode	Active Clock Domains					Oscillators		Wake-up Sources							Software BOD Disable
	clk _{CPU}	clk _{FLASH}	clk _{IO}	clk _{ADC}	clk _{ASY}	Main Clock Source Enabled	Timer Oscillator Enabled	INT1, INT0 and Pin Change	TWI Address Match	Timer2	SPM/EEPROM Ready	ADC	WDT	Other/O	
Idle			X	X	X	X	X ⁽²⁾	X	X	X	X	X	X	X	
ADC noise Reduction				X	X	X	X ⁽²⁾	X ⁽³⁾	X	X ⁽²⁾	X	X	X		
Power-down								X ⁽³⁾	X				X		X
Power-save					X		X ⁽²⁾	X ⁽³⁾	X	X			X		X
Standby ⁽¹⁾						X		X ⁽³⁾	X				X		X
Extended Standby					X ⁽²⁾	X	X ⁽²⁾	X ⁽³⁾	X	X			X		X

Tryby oszczędzania energii

Atmega328P - Idle

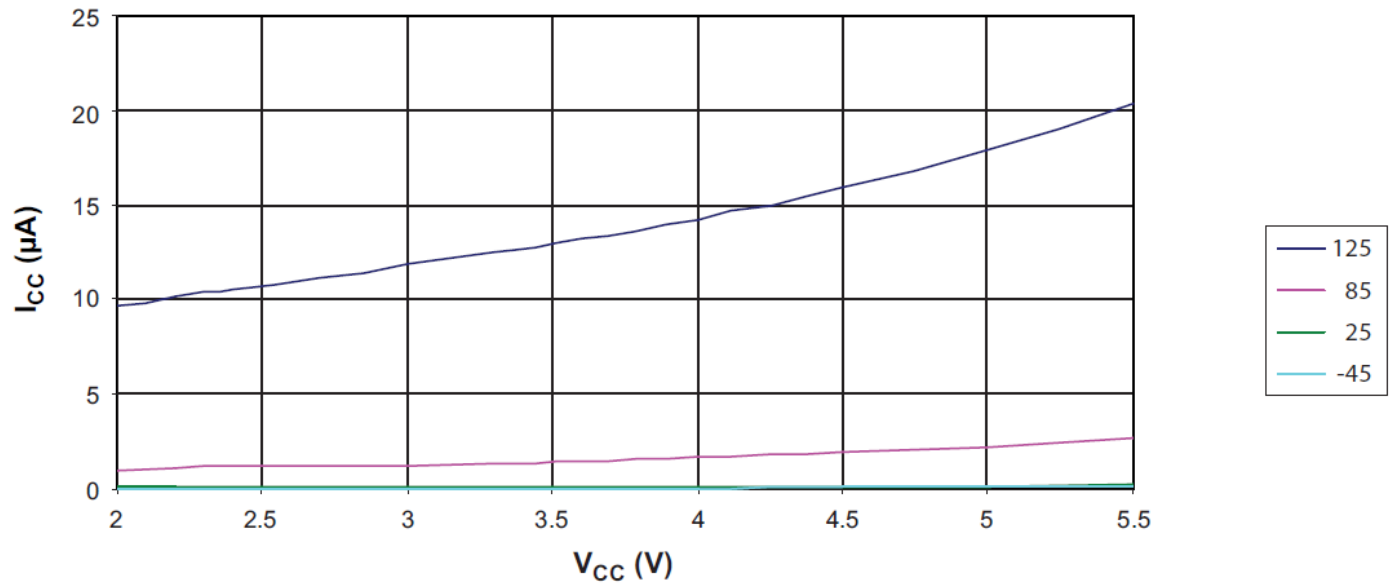
Figure 29-2. Idle Supply Current versus Frequency



Tryby oszczędzania energii

Atmega328P – Power Down

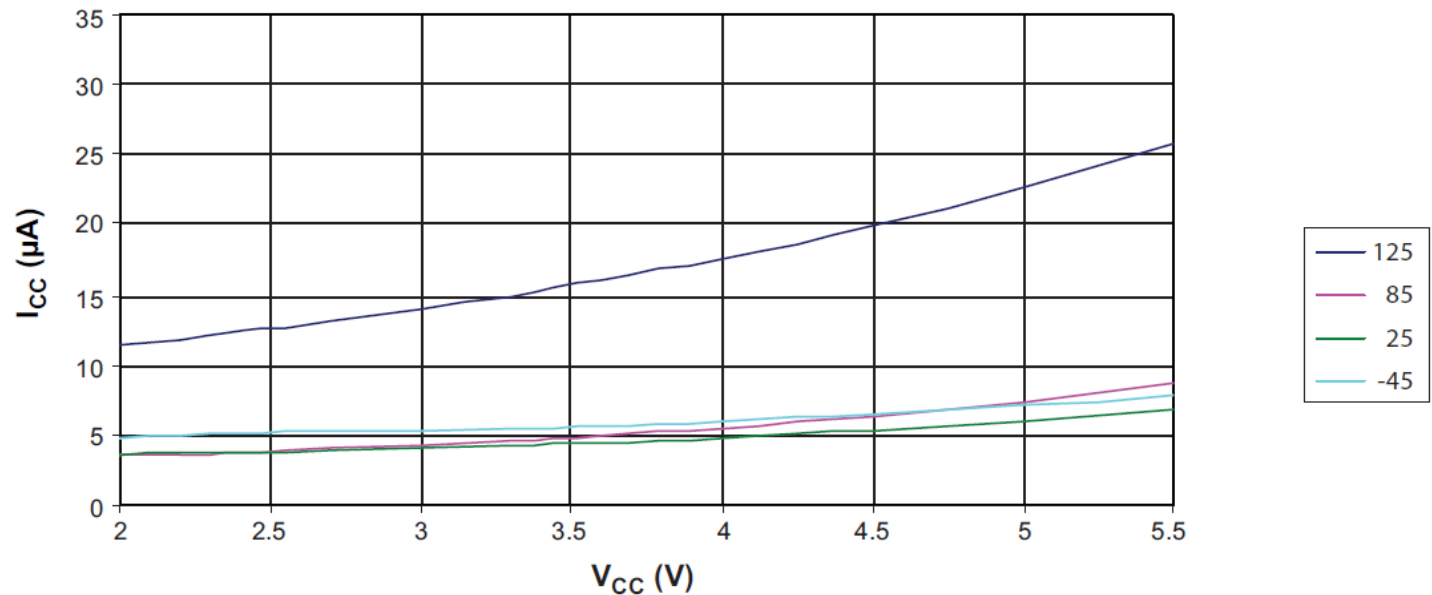
Figure 29-3. Power-Down Supply Current versus V_{CC} (Watchdog Timer Disabled)



Tryby oszczędzania energii

Atmega328P – Power Down

Figure 29-4. Power-Down Supply Current versus V_{CC} (Watchdog Timer Enabled)



Tryby oszczędzania energii

Atmega328P vs Arduino

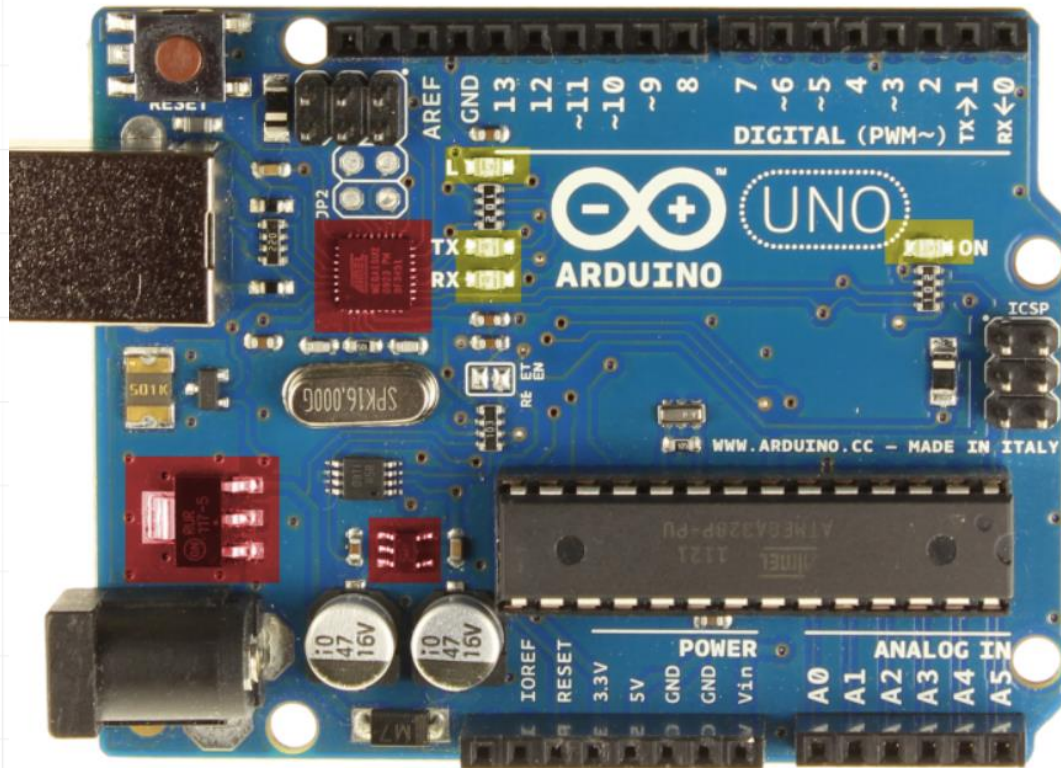
czerwony - peryferia włączone
zielony – peryferia wyłączone

Tryb	Arduino UNO	ATMega328
Idle	45 mA	15,2 mA
	56,4 mA	17 mA
ADC Noise Reduction	47,2 mA	5,8 mA
	44,8 mA	6 mA
Power-down	36,8 mA	6,5 uA
	37 mA	144,5 uA
Power-save	36,8 mA	6,5 uA
	38 mA	1,4 mA
Standby	37,2 mA	514 uA
	37,4 mA	650 uA
Extended Standby	37,3 mA	511 uA
	38,1 mA	1,4 mA

Tryby oszczędzania energii

Atmega328P vs Arduino

czerwony - peryferia włączone
zielony – peryferia wyłączone



Tryby oszczędzania energii

Atmega328P vs Arduino

Program typu „Blink”

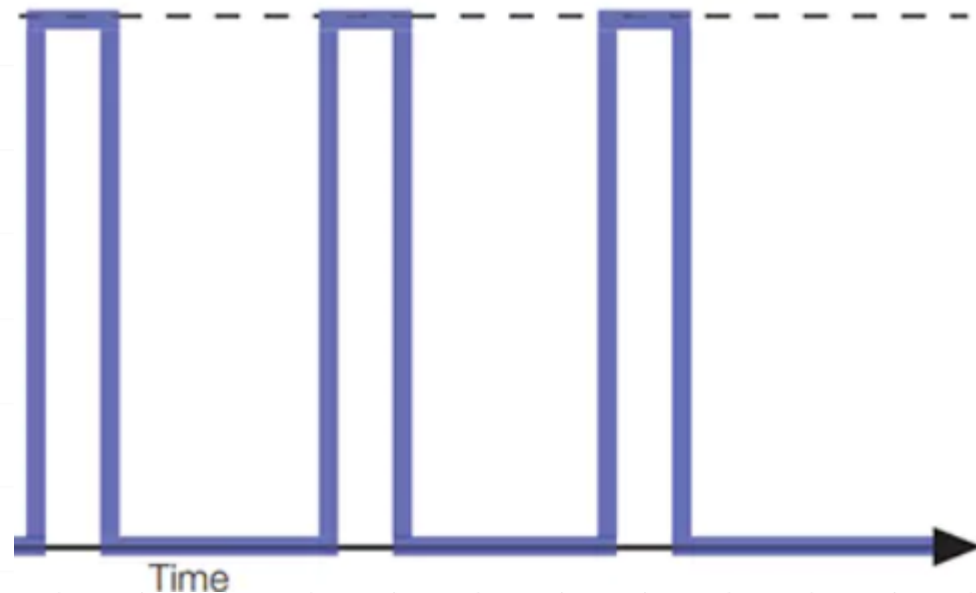
	Arduino UNO	ATMega328
Średnia wartość poboru prądu	53 mA	20,15 mA
Pojemność baterii	1500 mAh	1500 mAh
Czas pracy na baterii*	28,3 godziny	74,4 godziny (2,6 raza dłużej)

Tryby oszczędzania energii

Tryby oszczędzania energii – jak to stosować?

Program okresowy:

- wykonujemy potrzebne zadania jak najszybciej,
- przechodzimy do trybu oszczędzania energii,
- oczekujemy na kolejne wybudzenie
- wykonujemy....

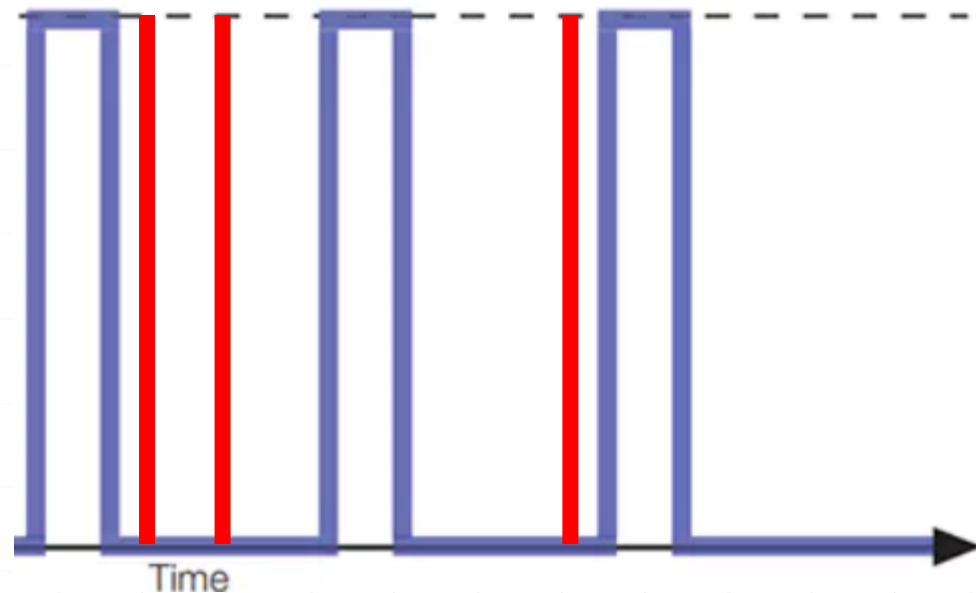


Tryby oszczędzania energii

Tryby oszczędzania energii – jak to stosować?

Bardziej rozbudowana aplikacja:

- okresowe pomiary
- zdarzenia zewnętrzne, przerwania
- komunikacja przychodząca



Tryby oszczędzania energii

Tryby oszczędzania energii – jak to stosować?

Zalety:

- oszczędzanie energii
- dłuższy czas pracy na zasilaniu bateryjnym

Wady:

- komplikacja programu
- zmiana podejścia przy programowaniu
- nie zawsze jest to możliwe
- podejście kompleksowe - program i elektronika wymaga odpowiedniego projektu

Tryby oszczędzania energii

Obliczanie czasu pracy na baterii

Idealny stan:

- pobór prądu: 50mA
- pojemność baterii: 500mAh
- ile będzie pracował układ na takim zasilaniu?

Rzeczywisty stan:

- pobór prądu jest dynamiczny, jak go określić dokładnie? Miganie diody LED? Komunikacja z układem peryferyjnym? Wykorzystanie okresowe zasobów wewnętrznych mikrokontrolera itd..
- Pojemność baterii – podana przez producenta? Zmierzona przy ładowaniu? Sprawność ogniwa? Efekt samorozładowania? Poziom bezpiecznego rozładowania/żywności?

Tryby oszczędzania energii

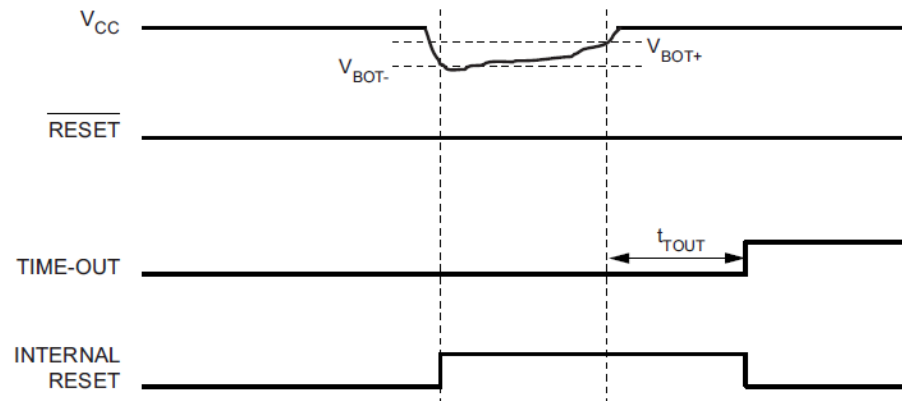
Atmega328P -BOD

10.5 Brown-out Detection

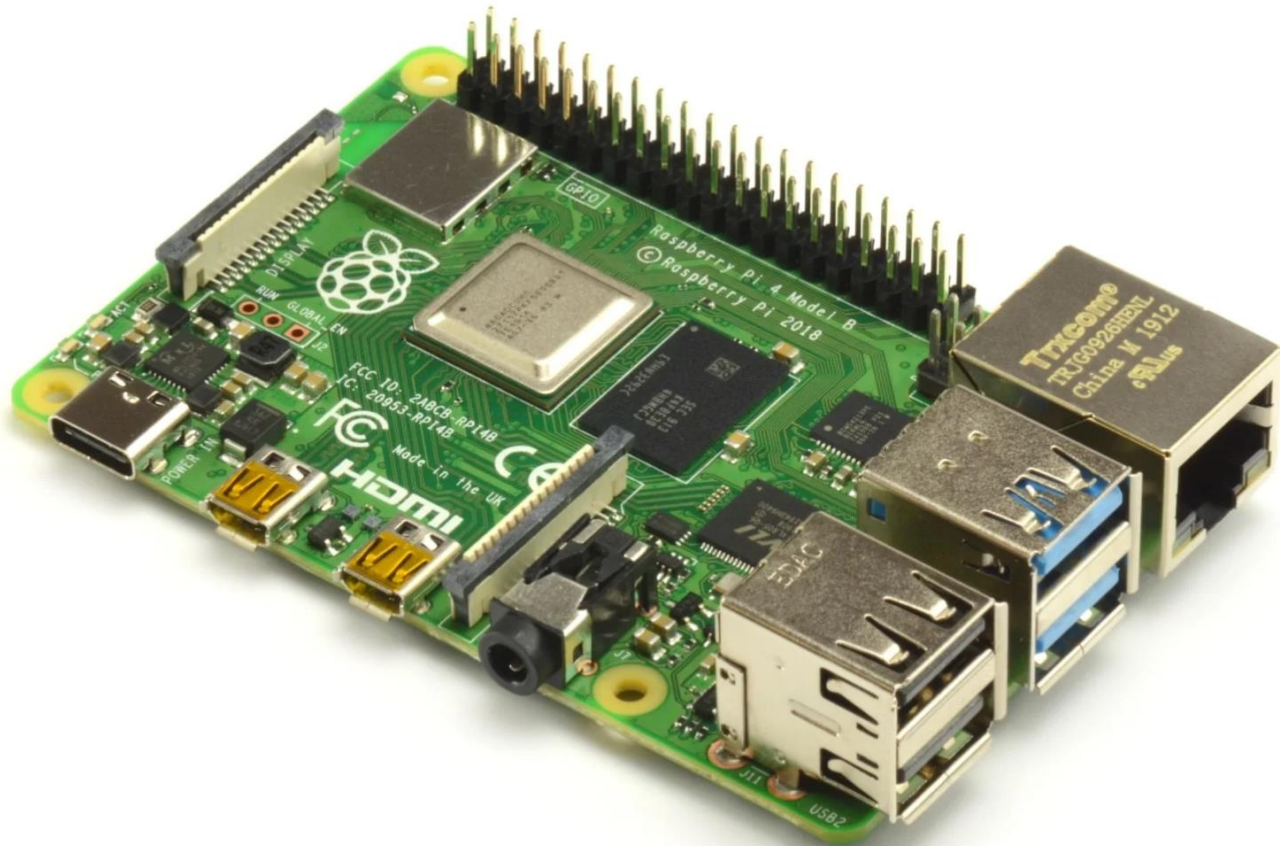
Atmel® ATmega328P has an on-chip brown-out detection (BOD) circuit for monitoring the V_{CC} level during operation by comparing it to a fixed trigger level. The trigger level for the BOD can be selected by the BODLEVEL fuses. The trigger level has a hysteresis to ensure spike free brown-out detection. The hysteresis on the detection level should be interpreted as $V_{BOT+} = V_{BOT} + V_{HYST}/2$ and $V_{BOT-} = V_{BOT} - V_{HYST}/2$. When the BOD is enabled, and V_{CC} decreases to a value below the trigger level (V_{BOT-} in Figure 10-5), the brown-out reset is immediately activated. When V_{CC} increases above the trigger level (V_{BOT+} in Figure 10-5), the delay counter starts the MCU after the time-out period t_{TOUT} has expired.

The BOD circuit will only detect a drop in V_{CC} if the voltage stays below the trigger level for longer than t_{BOD} given in Section 28.6 “System and Reset Characteristics” on page 261.

Figure 10-5. Brown-out Reset During Operation



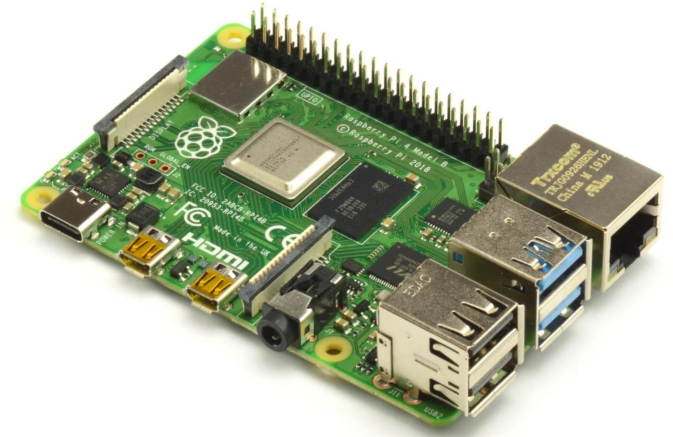
Tryby oszczędzania energii



Tryby oszczędzania energii

Raspberry Pi 4:

- napięcie zasilania: 5V
- pobór prądu maksymalny: 3A



Raspberry Pi 4 B

Pi State	Power Consumption
Idle	540 mA (2.7 W)
ab -n 100 -c 10 (uncached)	1010 mA (5.1 W)
400% CPU load (stress --cpu 4)	1280 mA (6.4 W)

Tryby oszczędzania energii

Raspberry Pi 4 B

Pi State	Power Consumption
Idle	540 mA (2.7 W)
ab -n 100 -c 10 (uncached)	1010 mA (5.1 W)
400% CPU load (stress --cpu 4)	1280 mA (6.4 W)

		1kWh - 1zł	
	Moc [W]	Rocznie [kWh]	Koszt roczny
Idle	2,70	23,65	23,65 zł
ab ...	5,10	44,68	44,68 zł
400% CPU load	6,40	56,06	56,06 zł

Tryby oszczędzania energii

Atmega328P vs Raspberry Pi 4

Raspberry PI 4, wysyłanie co 5 minut					
	Pobierany prąd [mA]	Czas trwania na okres [s]	Zużyta prąd na jedna próbkę [mAh]	Na dobę [mAh]	Na rok [mAh]
pomiar i wysyłanie	540	300	45,000000	12960,000	4730400,0
bezczyność	0	0	0,000000	0,000	0,000
		Razem:		12960,000	4730400,0

Atmega328P + moduł Lora, wysyłanie co 5 minut					
	Pobierany prąd [mA]	Czas trwania na okres [s]	Zużyta prąd na jedna próbkę [mAh]	Na dobę [mAh]	Na rok [mAh]
pomiar i wysyłanie	30	5	0,041667	12,000	4380,000
bezczyność	0,0065	290	0,000524	0,151	55,042
		Razem:		12,151	4435,042

?