



Elektronika cyfrowa

Moduł 2

Dr inż.
Patryk Król

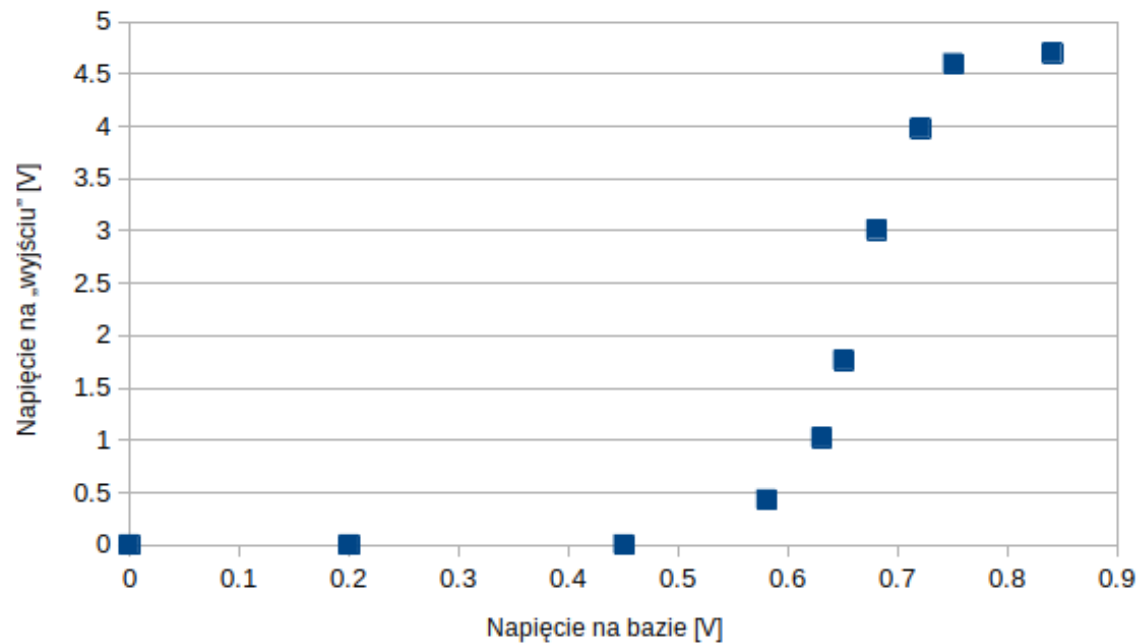
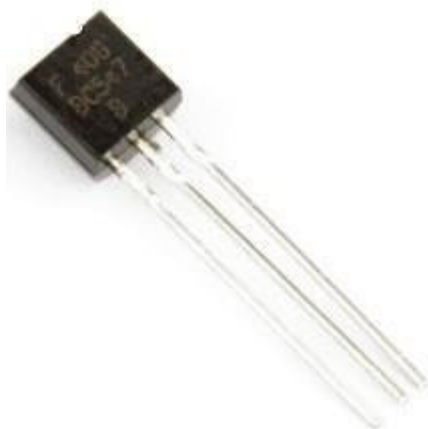
v2024/2



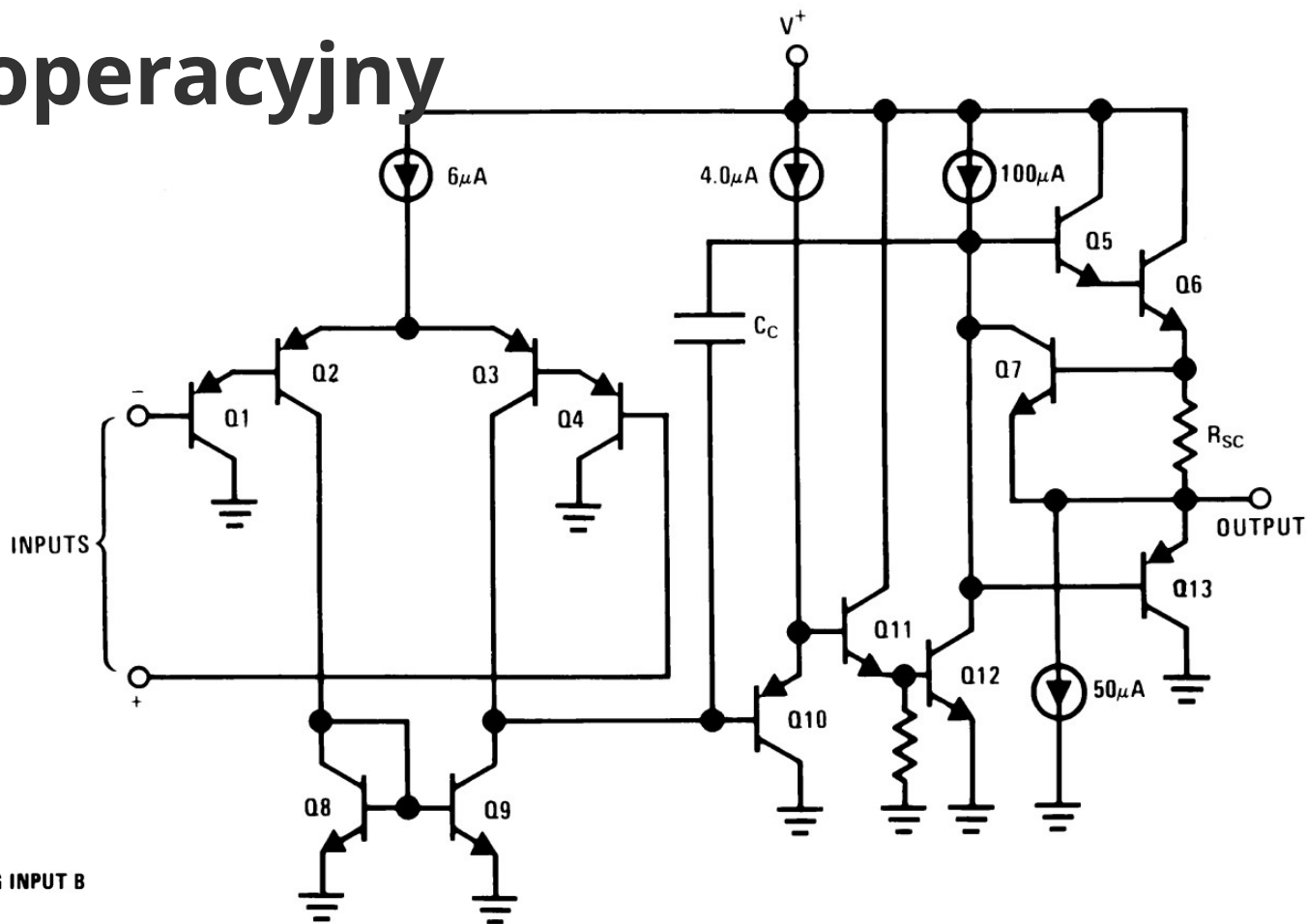
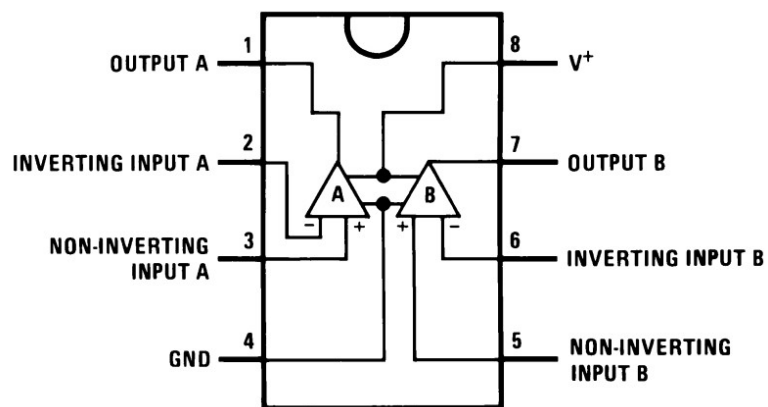
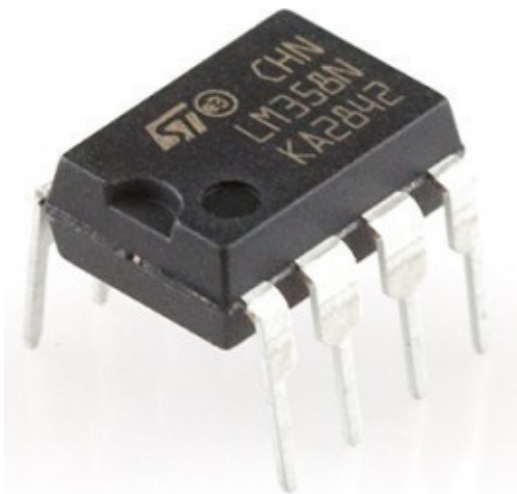
Moduł 2

- Tranzystor jako wzmacniacz
- Logika
- Elektronika cyfrowa
- TTL
- Przykładowe cyfrowe układy scalone:
przerzutniki, przetworniki C/A oraz A/C, bramki logiczne
- Mikrokontrolery

Tranzystor jako wzmacniacz

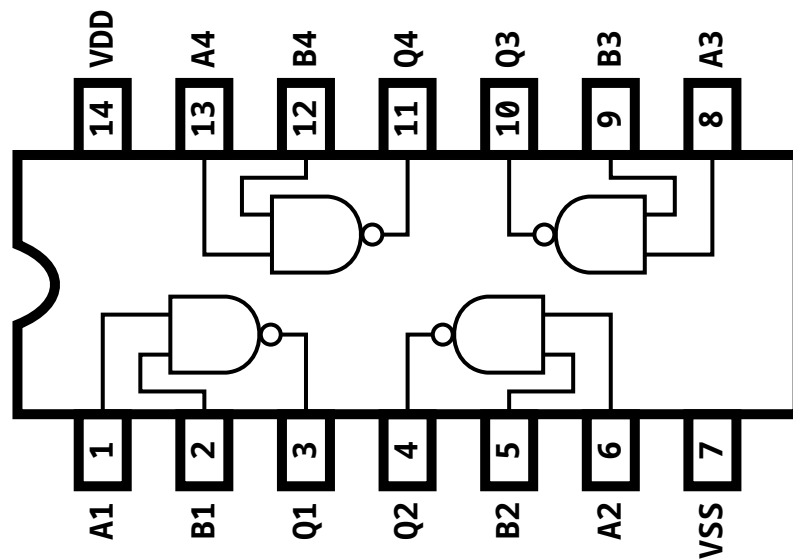


Wzmacniacz operacyjny

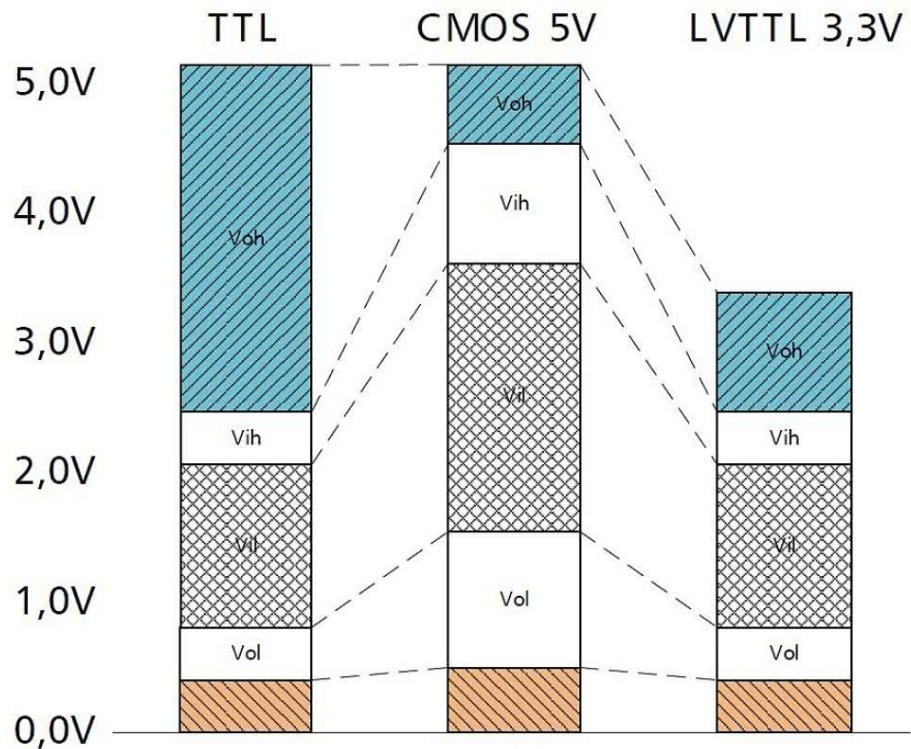


Logika

Logika (gr. λόγος, logos, rozum, słowo, myśl) – nauka o sposobach jasnego i ścisłego formułowania myśli, o regułach poprawnego rozumowania i uzasadniania twierdzeń.

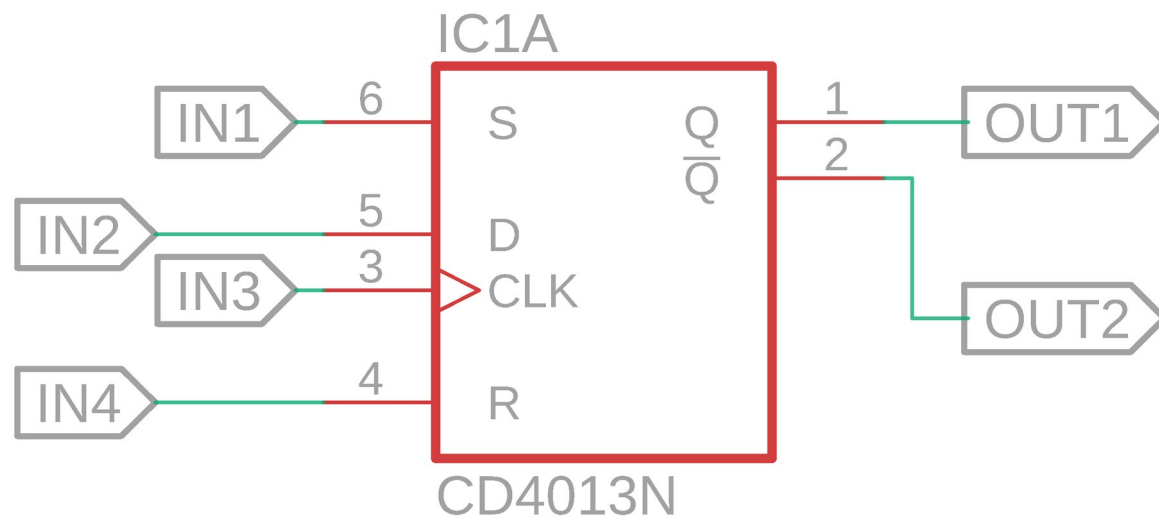
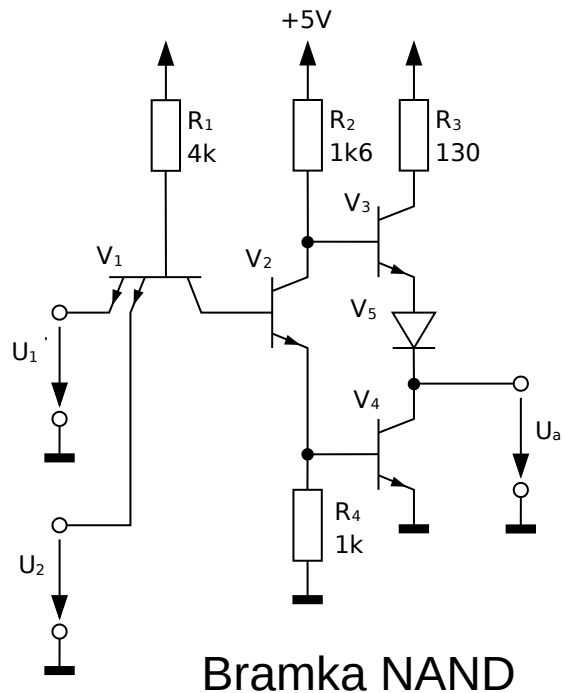


Transistor-transistor logic (TTL)

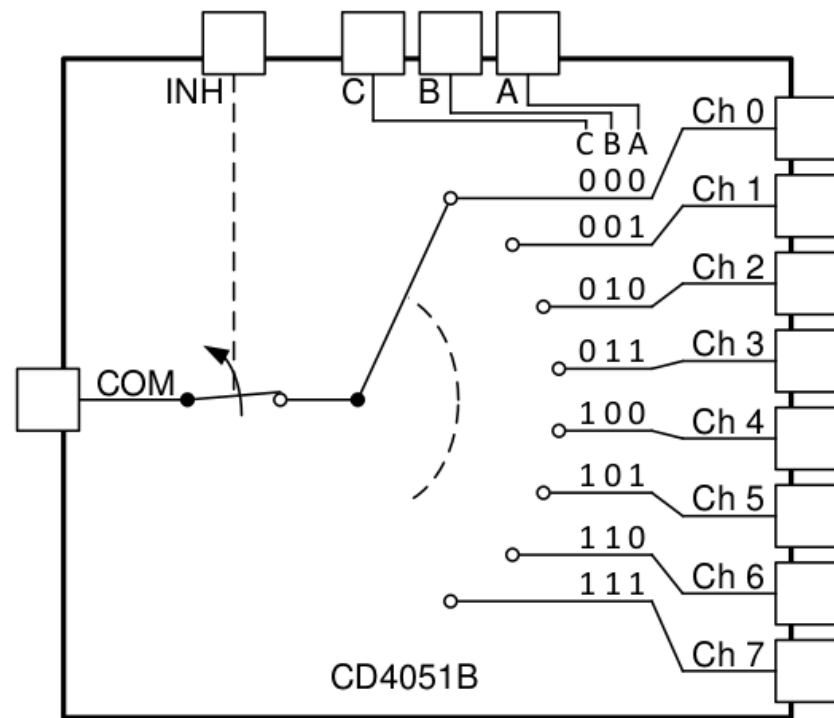
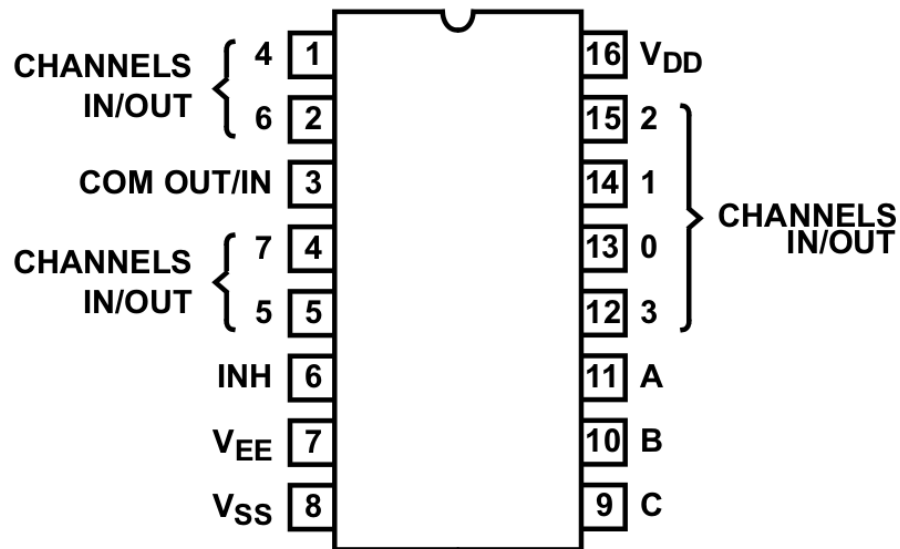


Napięcie [V]	Stan logiczny
0 – 0,8	Stan niski „0”, fałsz
0,8 – 2,4	Stan nieustalony
2,4 – 5	Stan wysoki „1”, prawda

Układy scalone



Multiplexer



Zapis binarny



$$N = b_n q^n \dots b_3 q^3 + b_2 q^2 + b_1 q^1 + b_0 q^0 + b_{-1} q^{-1} + b_{-2} q^{-2} \dots \text{etc.}$$

Zapis binarny do dziesiętnego

1	0	0	1	1	0	1	1
2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
128	64	32	16	8	4	2	1

wikiHow to Convert from Binary to Decimal

Zapis binarny do dziesiętnego

128 64 32 16 8 4 2 1

1 0 0 1 1 0 1 1

Zapis binarny do dziesiętnego

128	64	32	16	8	4	2	1
1	0	0	1	1	0	1	1
128	0	0	16	8	0	2	1

wikiHow to Convert from Binary to Decimal

Zapis binarny do dziesiętnego

128	64	32	16	8	4	2	1
1	0	0	1	1	0	1	1

$128 + 0 + 0 + 16 + 8 + 0 + 2 + 1$
 $= 155$

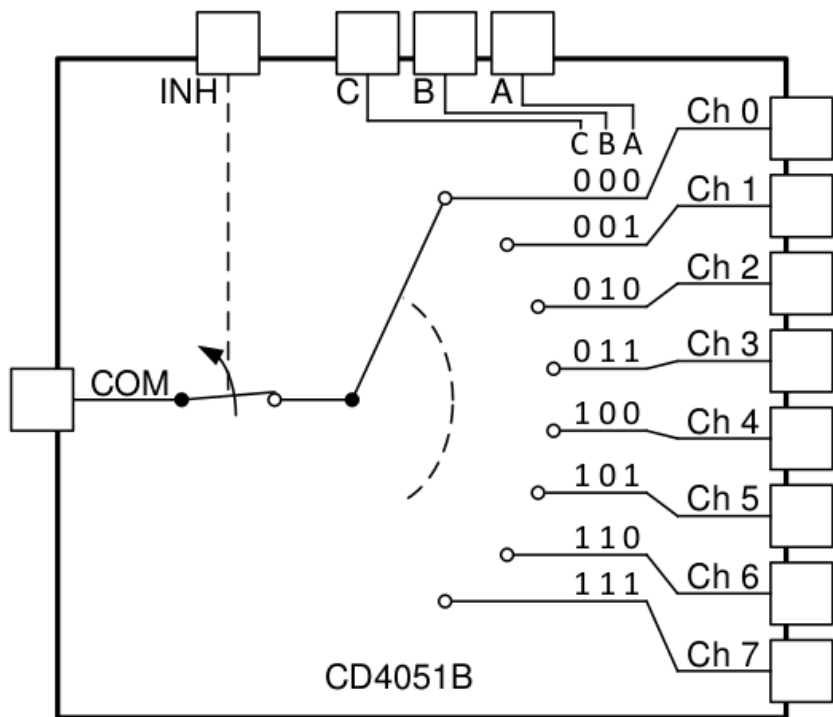
wikiHow to Convert from Binary to Decimal

Zapis binarny do dziesiętnego

10011011₂

= 155₁₀

Multiplexer



C	B	A

2 ²	2 ¹	2 ⁰
4	2	1

Tabela prawdy

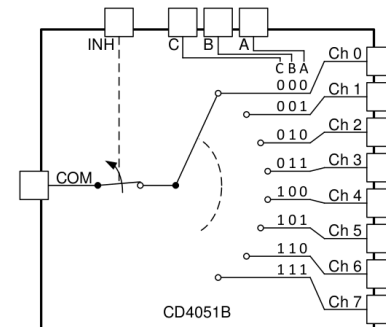


Table 1. Truth Table⁽¹⁾

INPUT STATES				ON CHANNEL(S)
INHIBIT	C	B	A	
CD4051B				
0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	X	X	X	None

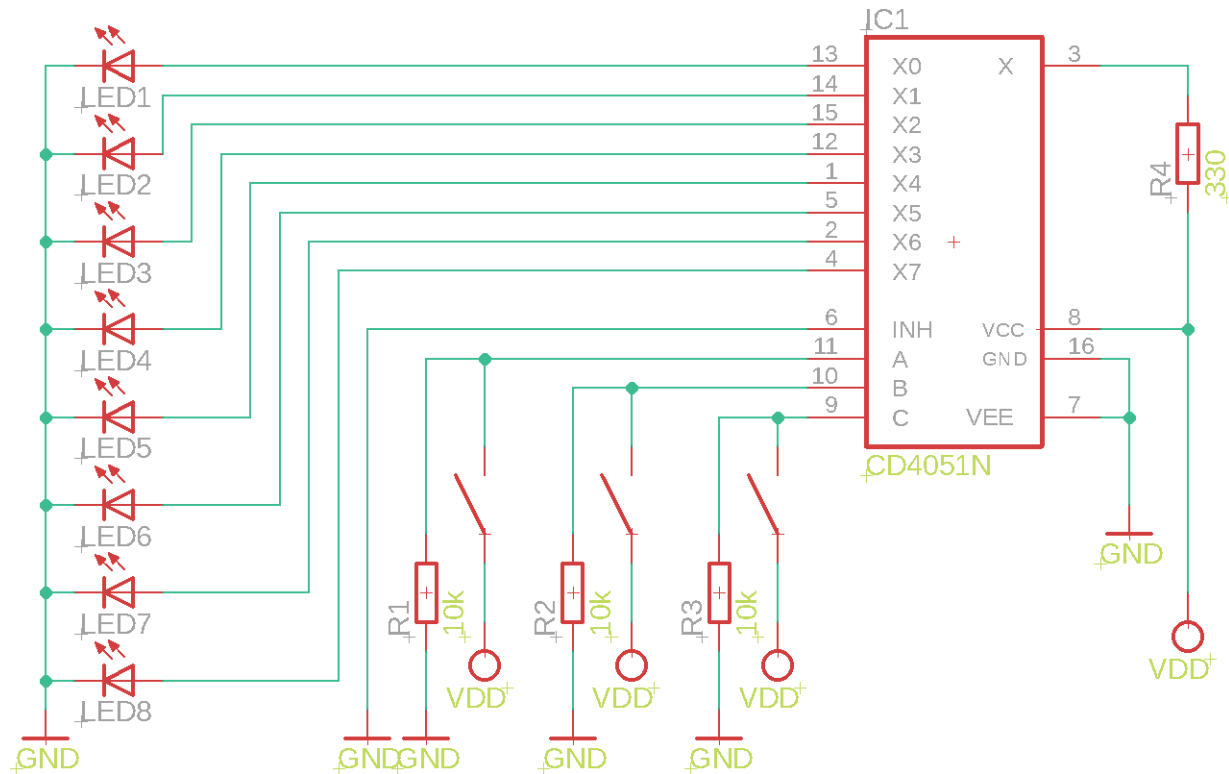
X – bez znaczenia

Multiplexer



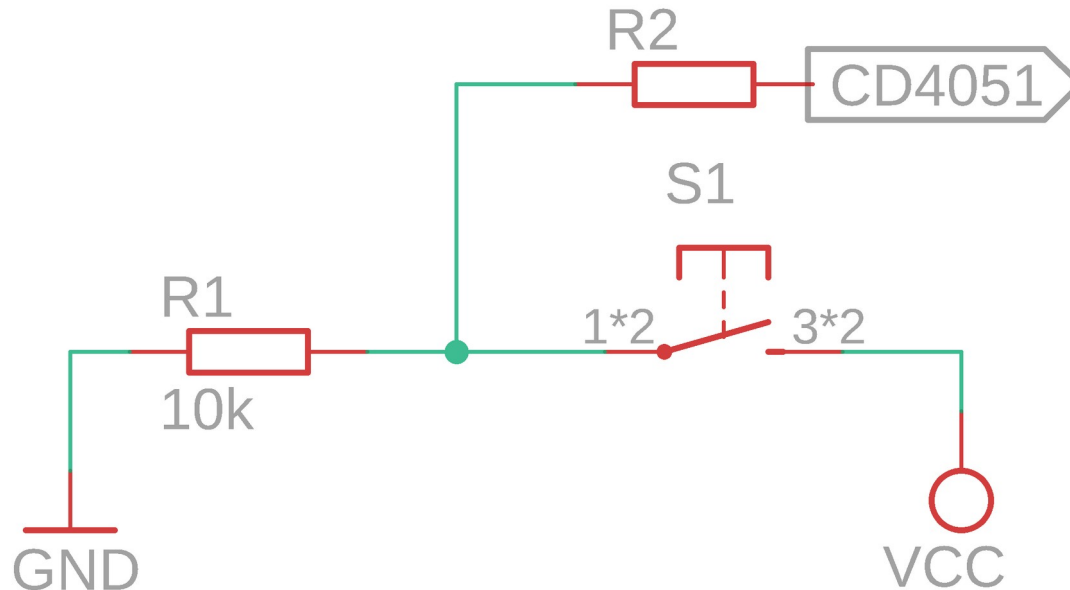
Table 1. Truth Table⁽¹⁾

INPUT STATES				ON CHANNEL(S)
INHIBIT	C	B	A	
CD4051B				
0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	X	X	X	None

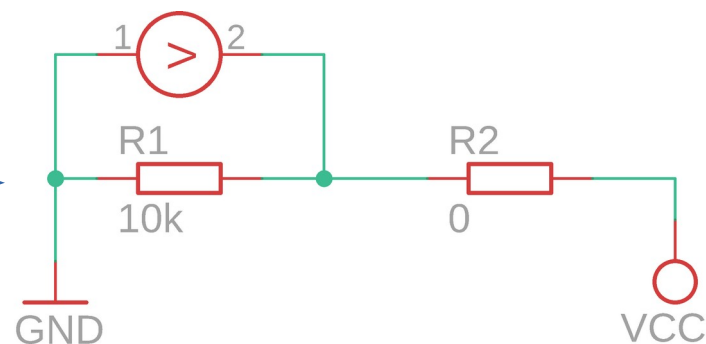
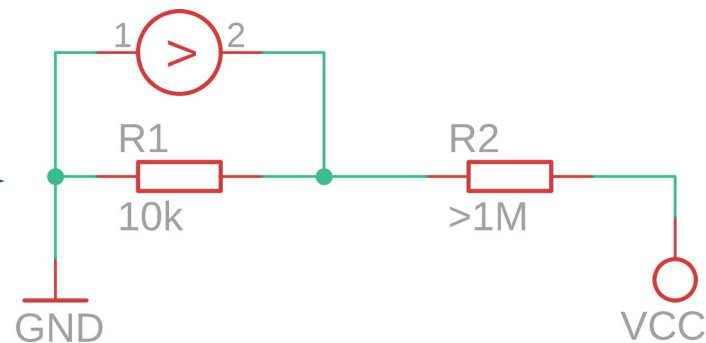
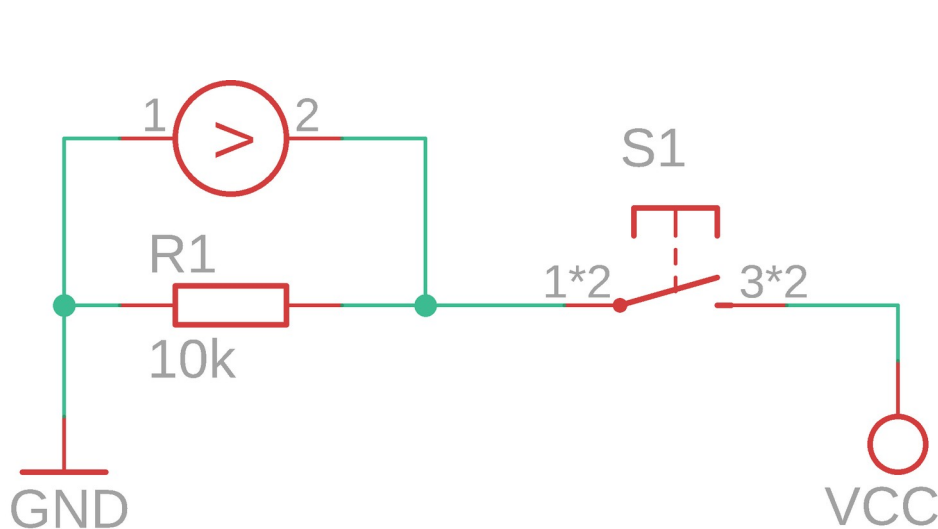


Dlaczego przycisk z rezystorem?

- 100% Tested for Quiescent Current at 20 V
- Maximum Input Current of 1 μA at 18 V Over Full Package Temperature Range, 100 nA at 18 V and 25°C
- Break-Before-Make Switching Eliminates Channel



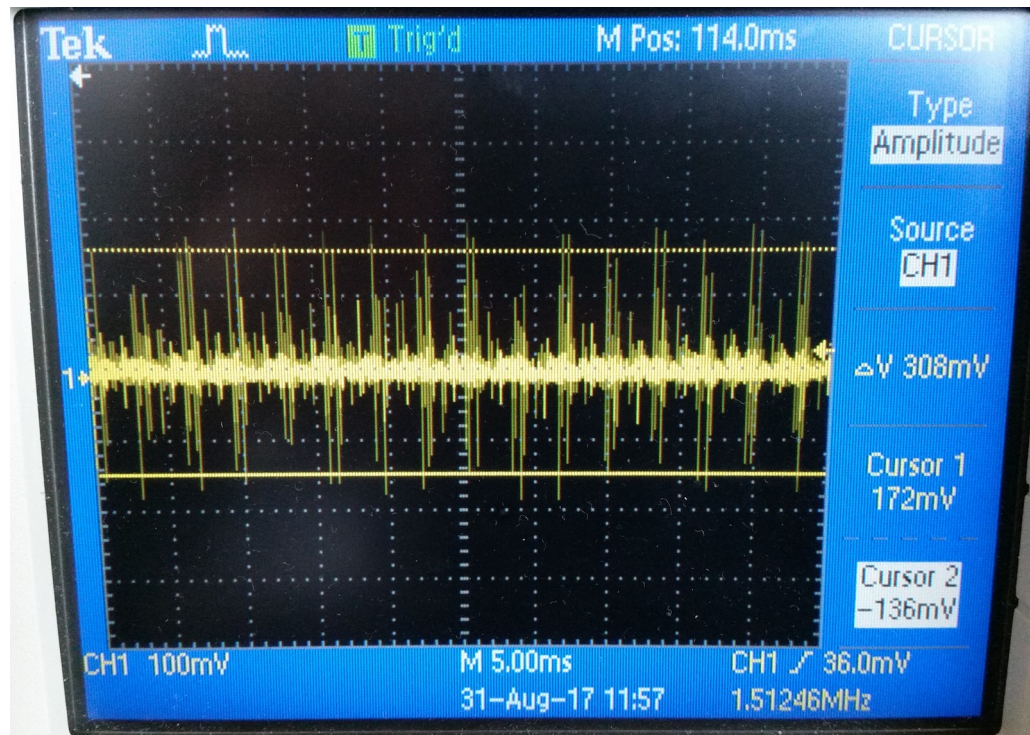
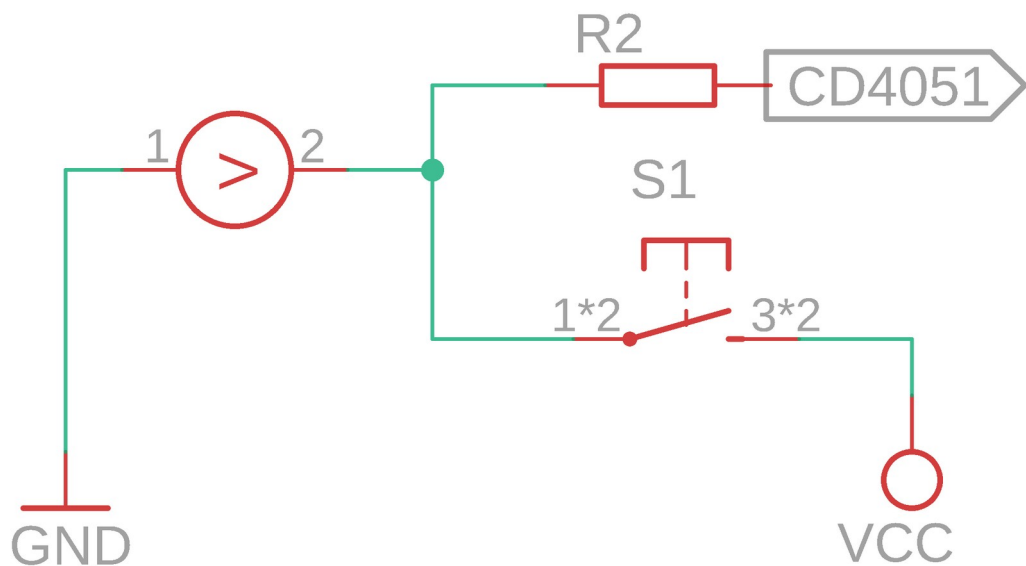
Dlaczego przycisk z rezystorem?



Input High Voltage, V_{IH} , Min

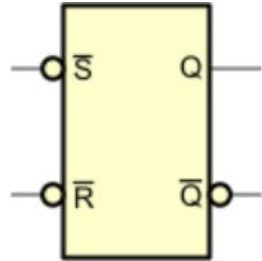
5	-55°C	3.5
	-40°C	3.5
	25°C	3.5
	85°C	3.5
	125°C	3.5

Dlaczego przycisk z rezystorem?

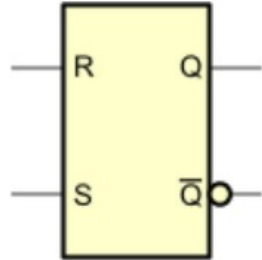


$$R_2 = \frac{18\text{ V}}{1 \cdot 10^{-6}\text{ A}} = 18\text{ M}$$

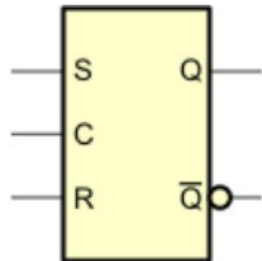
Przerzutnik



Przerzutnik S-R.
Wejścia aktywne w stanie niskim

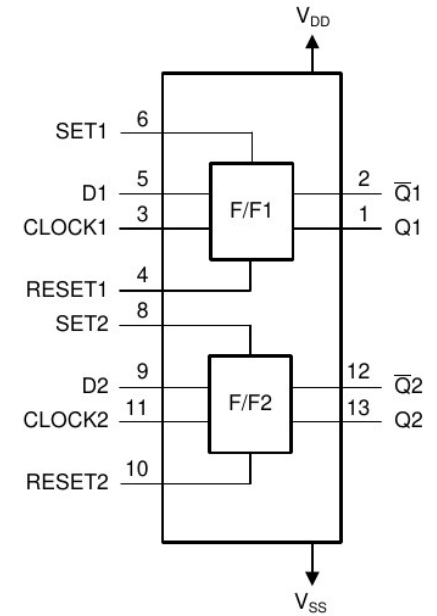


Przerzutnik R-S typu Latch.
Wejścia aktywne w stanie wysokim.



Synchroniczny przerzutnik S-R.
Wejścia aktywne w stanie wysokim przy wysokim stanie wejścia C.

CD4051



Copyright © 2016, Texas Instruments Incorporated

Table 1. Function Table

INPUTS				OUTPUT (Q)	INVERTED OUTPUT ($\bar{Q}$)
CLOCK	SET	RESET	D		
$\uparrow$	0	0	0	0	1
$\uparrow$	0	0	1	1	0
$\downarrow$	0	0	X	Q_0	$\bar{Q}$
X	0	1	X	0	1
X	1	0	X	1	0
X	1	1	X	1	1

CD4013

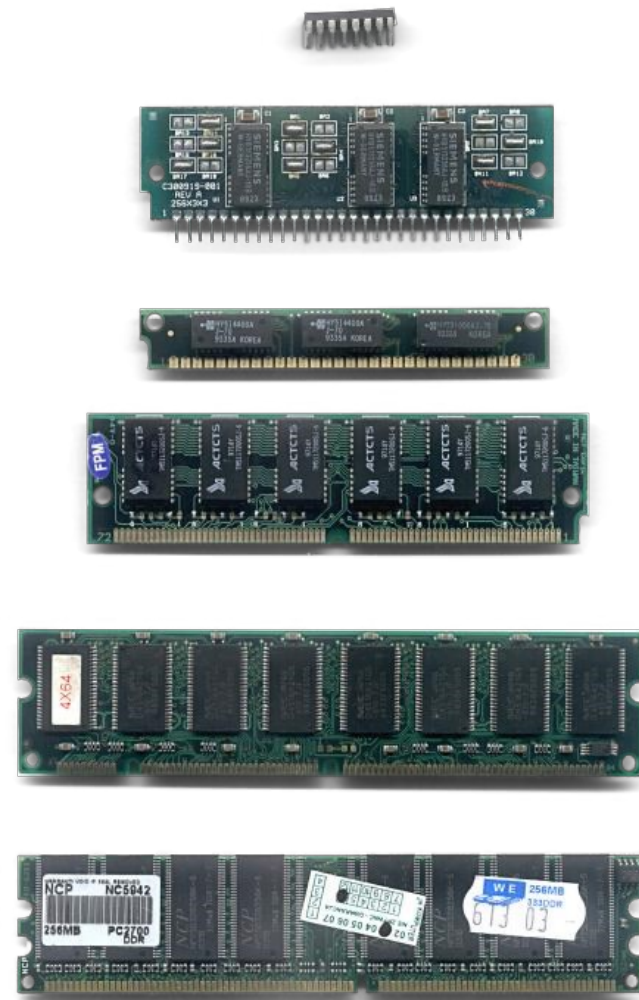
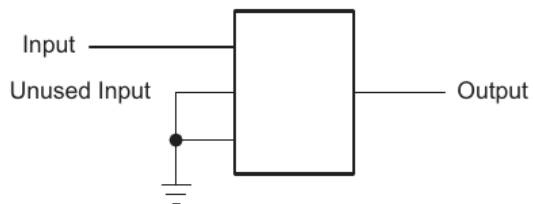
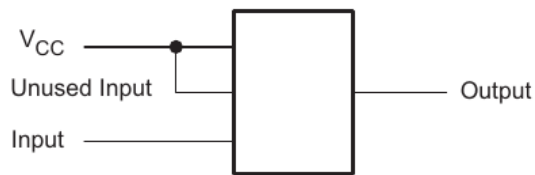
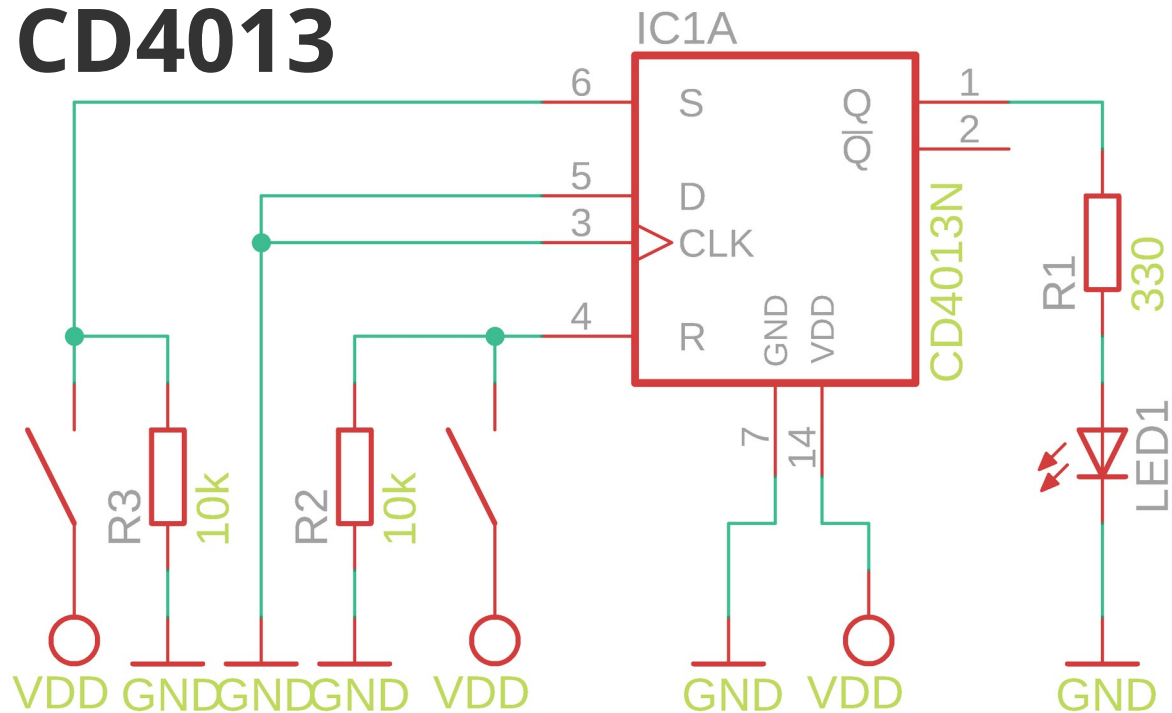
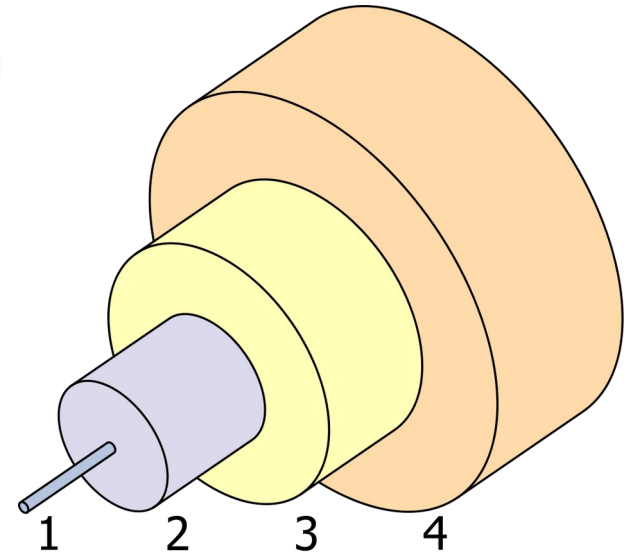
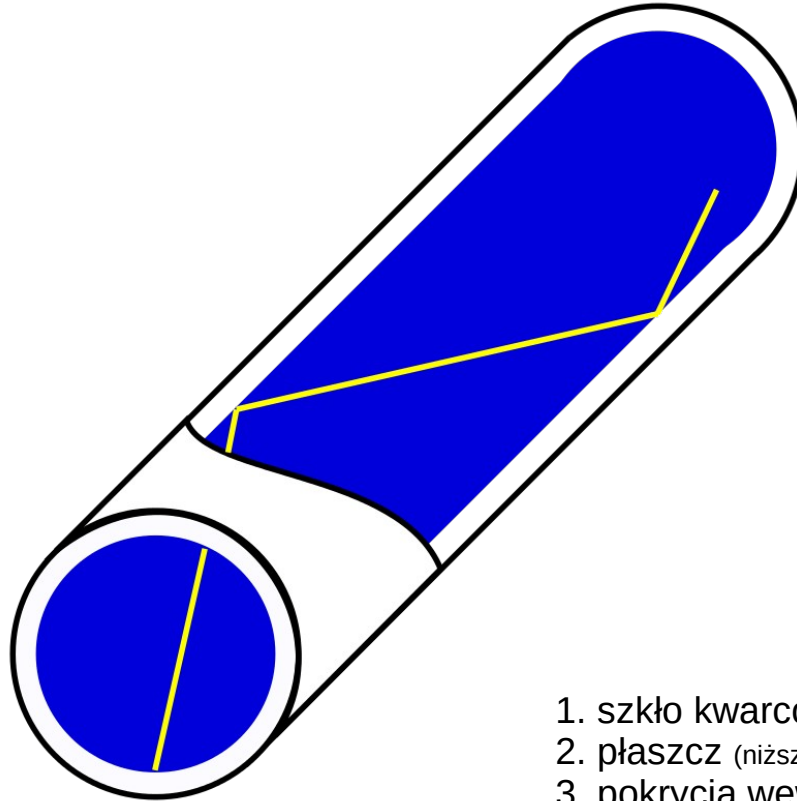
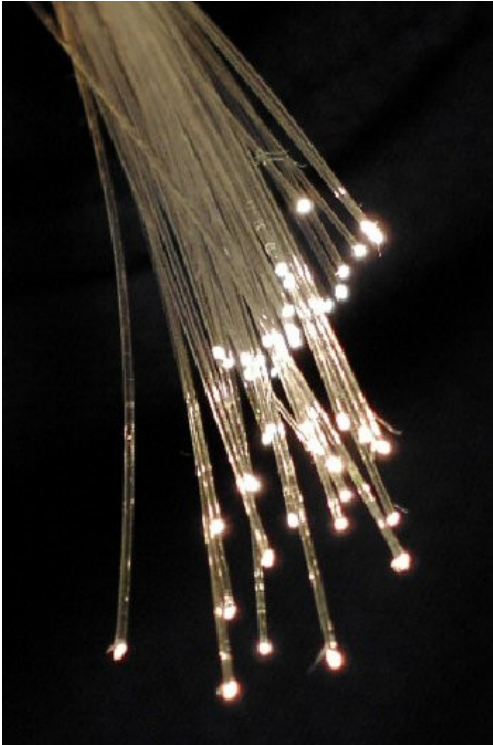


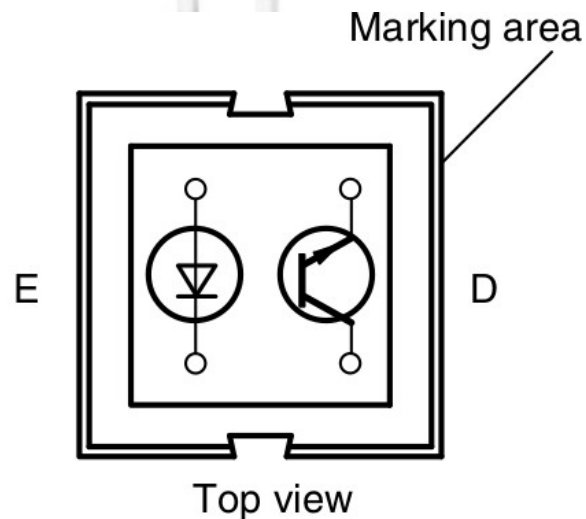
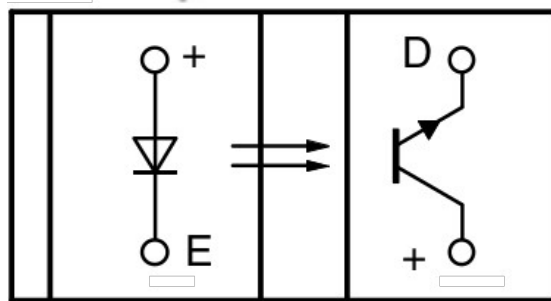
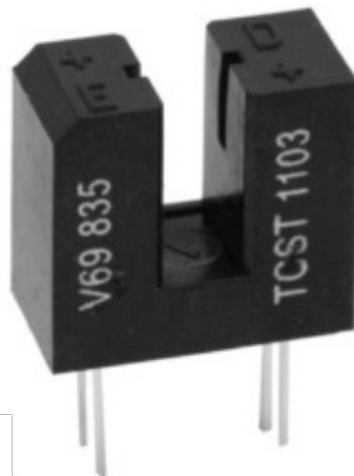
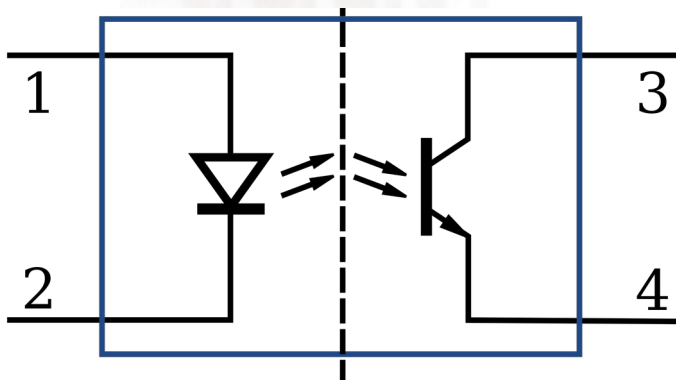
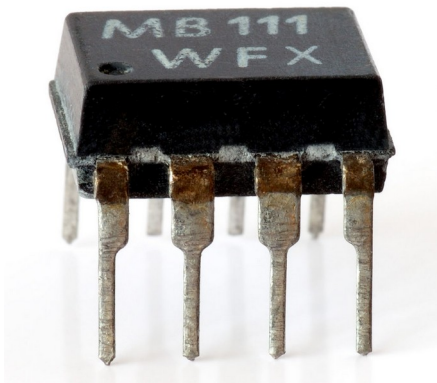
Figure 11. Layout Example for CD4013B

Inne metody transmisji danych (cyfrowych) - światłowód



1. szkło kwarcowe lub plastik : 8 μm
2. płaszcz (niższy współczynnik załamania światła): 125 μm
3. pokrycia wewnętrzne: 250 μm
4. pokrycie zewnętrzne: 400 μm

Optoizolator (transoptor)



Źródła:

Fairchild semiconductor, Phototransistor optical interrupter switch [6/05/01, DS300291]

VISHAY: Transmissive Optical Sensor with Phototransistor Output, Document Number: 83764, Rev. 1.9, 17-Aug-09

<https://en.wikipedia.org/wiki/Opto-isolator>

Vishay Semiconductors: CNY70

Przetwornik A/C

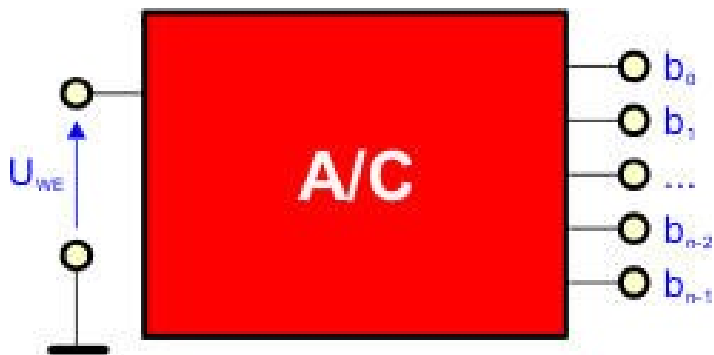
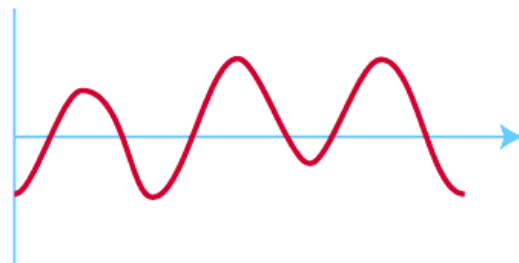
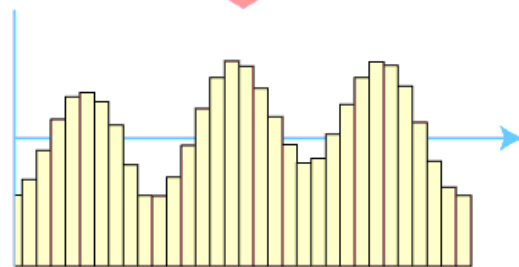


fig. 1



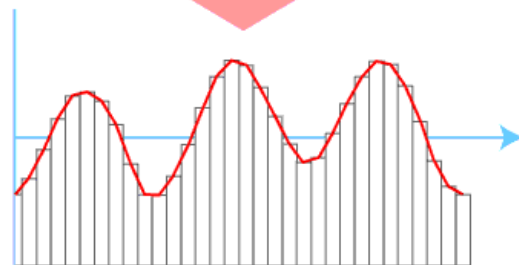
Conversion A-D

fig. 2



Conversion D-A

fig. 3



Dwójkowo	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
Dziesiętnie	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0...5V:	0V	0,3V	0,7V	1,0V	1,3V	1,7V	2,0V	2,3V	2,7V	3,0V	3,3V	3,7V	4,0V	4,3V	4,7V	5,0V

Źródła:

https://en.wikipedia.org/wiki/Analog-to-digital_converter

https://eduinf.waw.pl/inf/prg/009_kurs_avr/2017.php

Sygnal analogowy → cyfrowy



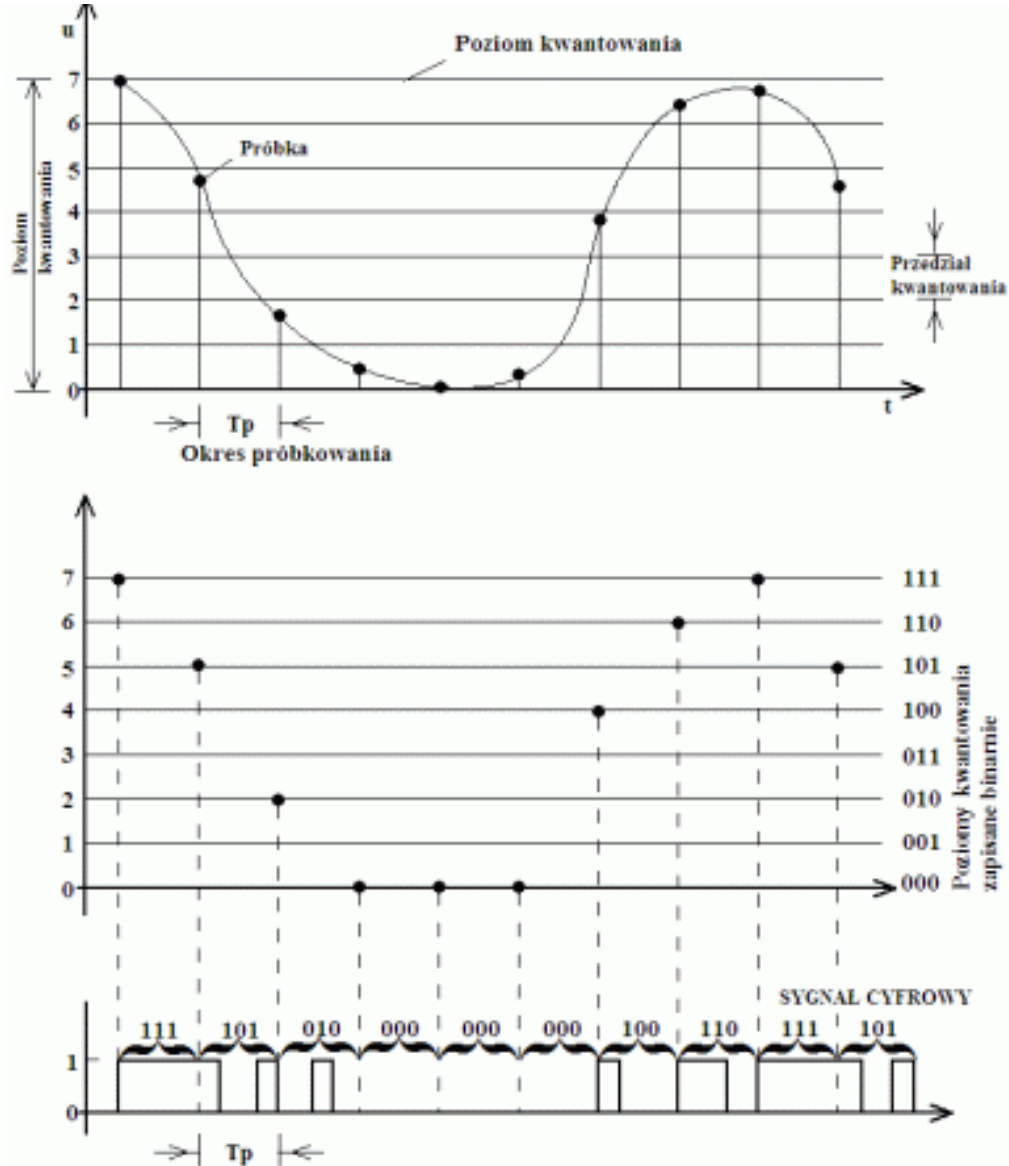
Przetworniki C/A



Przetworniki C/A



Próbkowanie i rozdzielczość bitowa



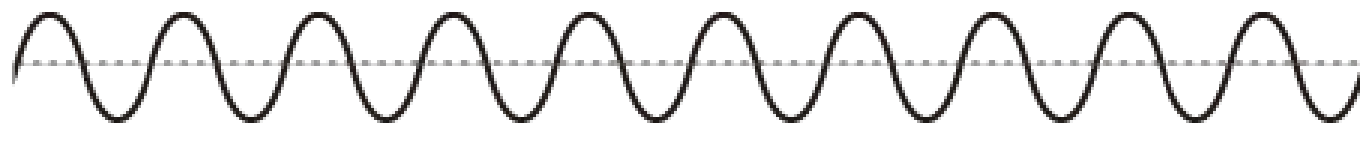
Próbkowanie
sygnału analogowego

Kwantowanie
próbek

Kodowanie
próbek

Próbkowanie i rozdzielczość bitowa

ORIGINAL AUDIO



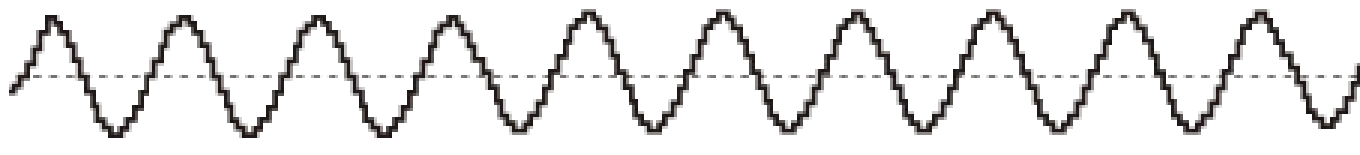
RECORDING
STUDIO

24 Bit HIGH
RESOLUTION AUDIO



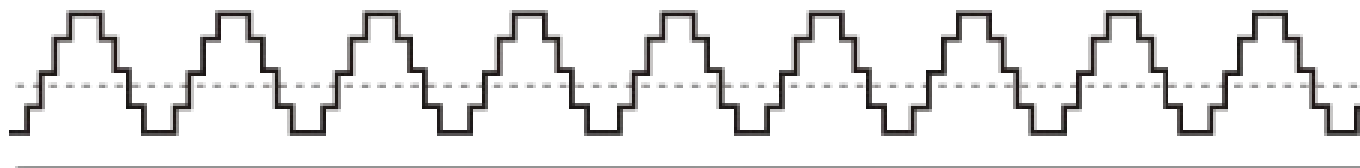
HIGH RESOLUTION
AUDIO PLAYER

16 Bit AUDIO



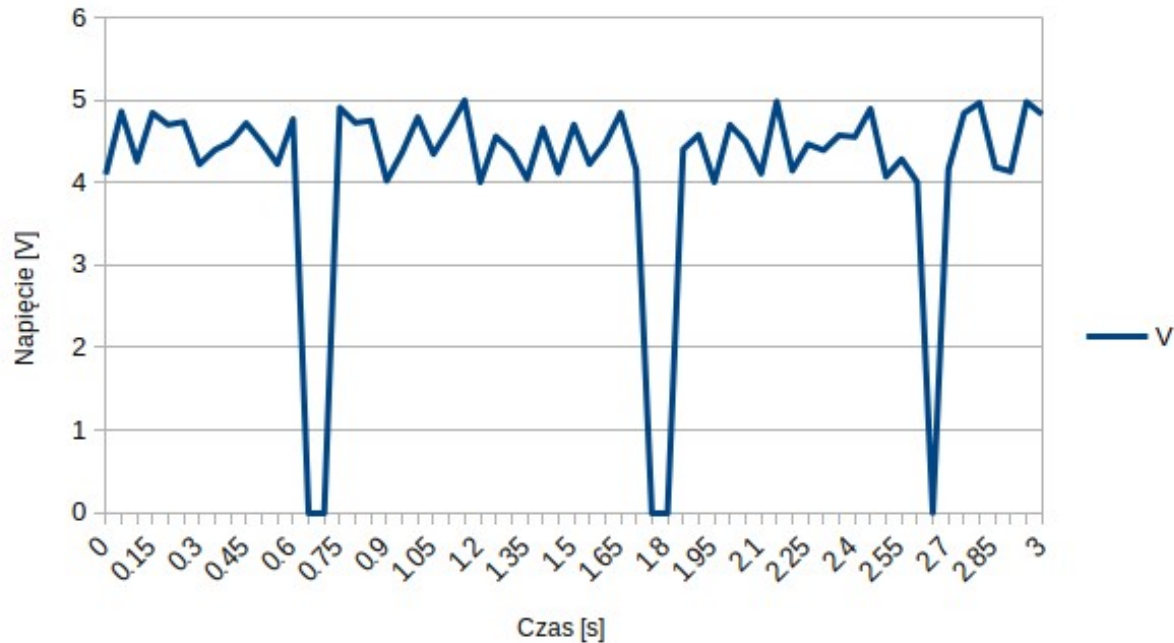
COMPACT DISC

MP3 AUDIO

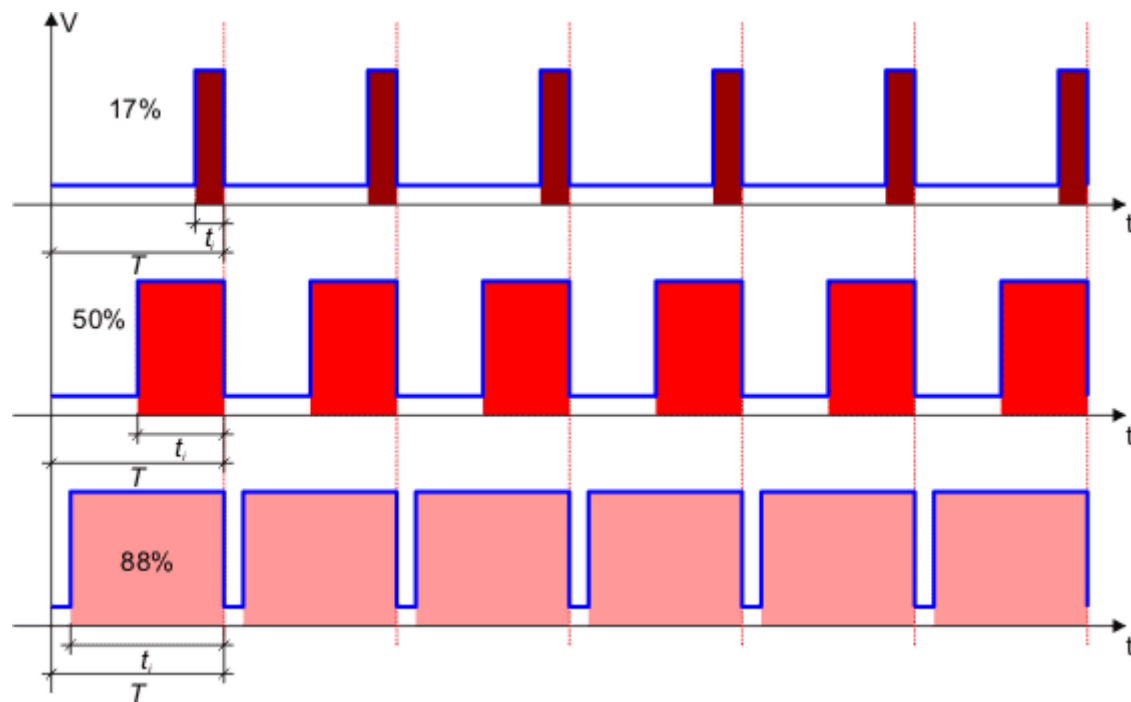


MP3 PLAYER

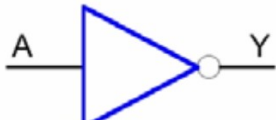
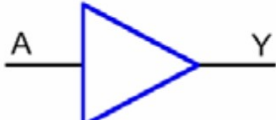
Przetworniki A/C



Przetwornik C/A, PWM



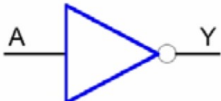
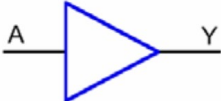
Inne układy cyfrowe – bramki logiczne

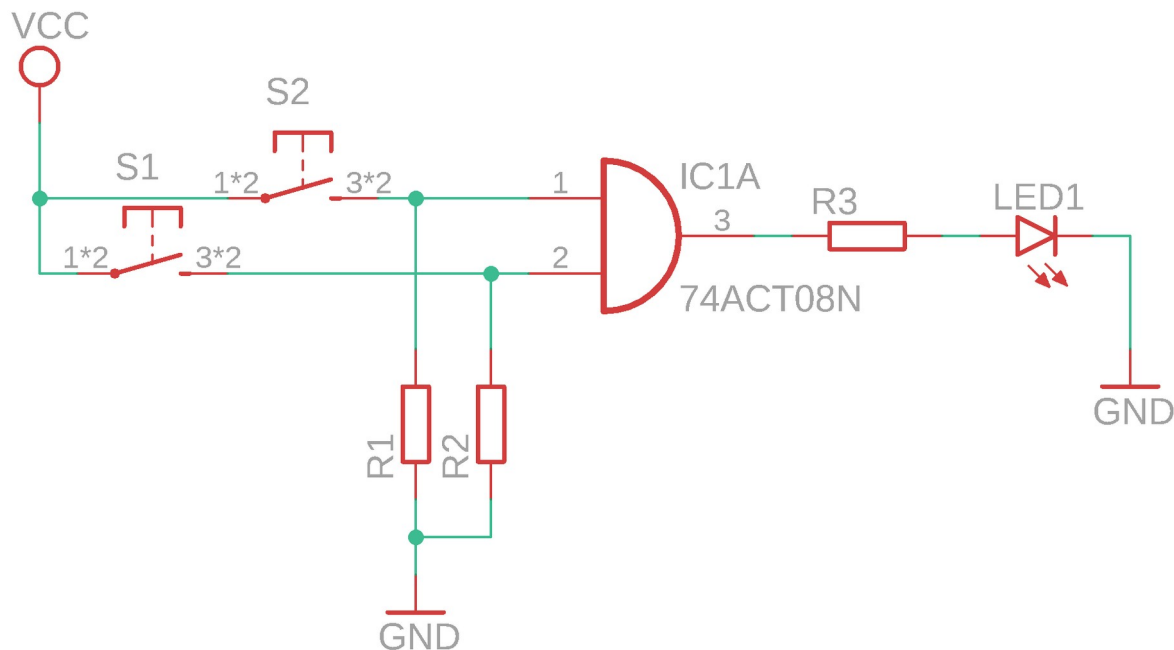
Rodzaj bramki	Symbol bramki	Tabela prawdy															
AND		<table><tr><th>A</th><th>B</th><th>$Y = A \cdot B$</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	A	B	$Y = A \cdot B$	0	0	0	0	1	0	1	0	0	1	1	1
A	B	$Y = A \cdot B$															
0	0	0															
0	1	0															
1	0	0															
1	1	1															
OR		<table><tr><th>A</th><th>B</th><th>$Y = A + B$</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	A	B	$Y = A + B$	0	0	0	0	1	1	1	0	1	1	1	1
A	B	$Y = A + B$															
0	0	0															
0	1	1															
1	0	1															
1	1	1															
INWERTER		<table><tr><th>A</th><th>$Y = \bar{A}$</th></tr><tr><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td></tr></table>	A	$Y = \bar{A}$	0	1	1	0									
A	$Y = \bar{A}$																
0	1																
1	0																
BUFOR		<table><tr><th>A</th><th>$Y = A$</th></tr><tr><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td></tr></table>	A	$Y = A$	0	0	1	1									
A	$Y = A$																
0	0																
1	1																



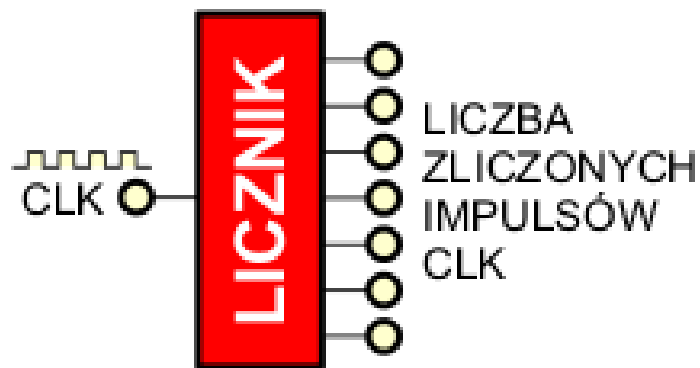
Źródło:
http://layer.uci.agh.edu.pl/~maglay/wrona/pl/podstrony/dydaktyka/Technika_Cyfrowa/Bramki/Bramki_teor.pdf

Inne układy cyfrowe – bramki logiczne

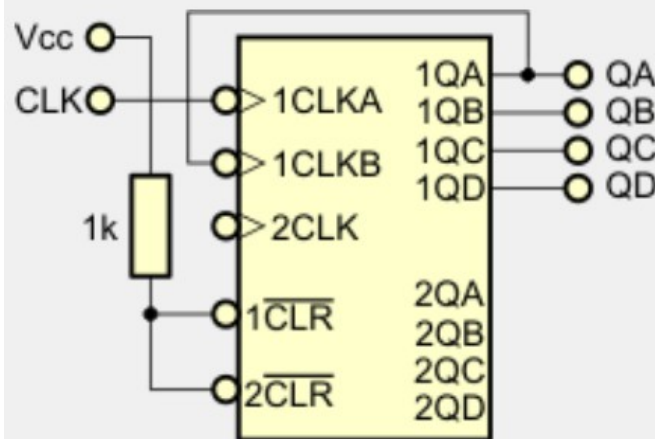
Rodzaj bramki	Symbol bramki	Tabela prawdy		
AND		A	B	$Y = A \cdot B$
		0	0	0
		0	1	0
		1	0	0
		1	1	1
OR		A	B	$Y = A + B$
		0	0	0
		0	1	1
		1	0	1
		1	1	1
INWERTER		A	$Y = \overline{A}$	
		0	1	
		1	0	
BUFOR		A	$Y = A$	
		0	0	
		1	1	



Inne układy cyfrowe - licznik

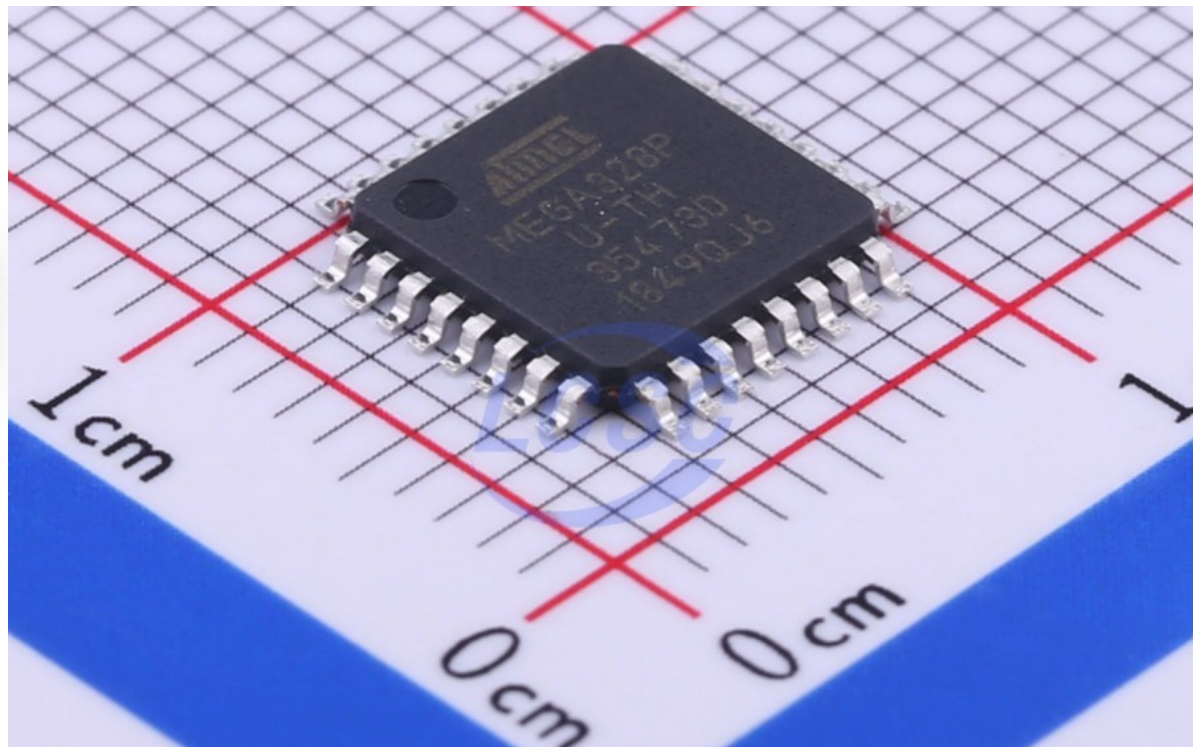


Tryb dziesiętny (oba liczniki)



n	QD	QC	QB	QA
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1

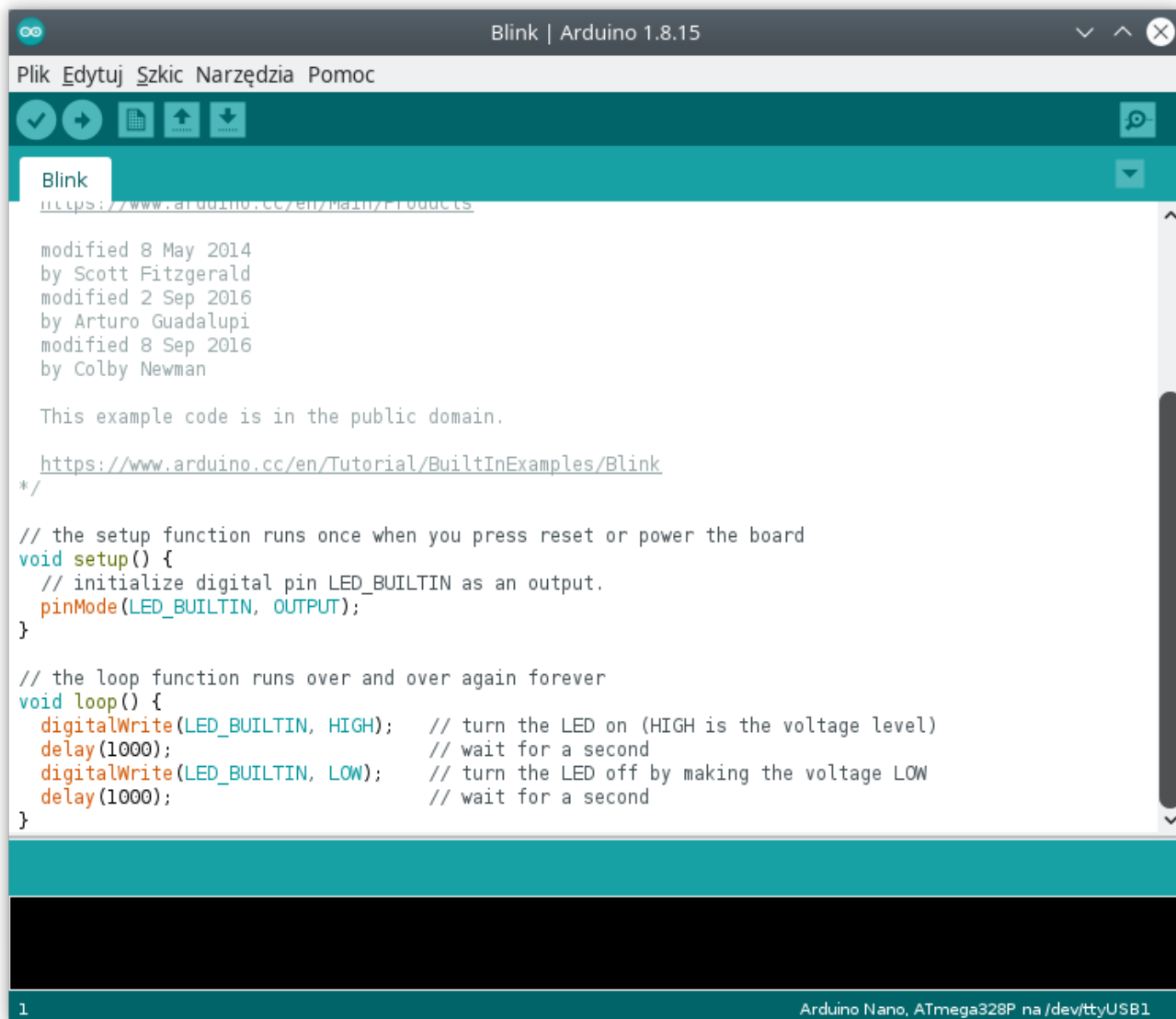
Inne układy cyfrowe - mikrokomputer



Flash 32 kB, RAM 2 kB, EEPROM 1 kB, A/D, 6x PWM, 1.8 do 5 V

37 / 39

C++



The screenshot shows the Arduino IDE interface. The title bar reads "Blink | Arduino 1.8.15". The menu bar includes "Plik", "Edytuj", "Szkic", "Narzędzia", and "Pomoc". The toolbar contains icons for saving, undo, redo, and uploading. The main text area displays the "Blink" example code, which is a simple program that toggles the built-in LED on and off every second. The code is as follows:

```
modified 8 May 2014
by Scott Fitzgerald
modified 2 Sep 2016
by Arturo Guadalupi
modified 8 Sep 2016
by Colby Newman

This example code is in the public domain.

https://www.arduino.cc/en/Tutorial/BuiltInExamples/Blink
*/

// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin LED_BUILTIN as an output.
  pinMode(LED_BUILTIN, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
  digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(1000);                     // wait for a second
  digitalWrite(LED_BUILTIN, LOW);  // turn the LED off by making the voltage LOW
  delay(1000);                     // wait for a second
}
```

The status bar at the bottom indicates "1" on the left and "Arduino Nano, ATmega328P na /dev/ttyUSB1" on the right.

Technologiczne odmiany układów cyfr.

Wyróżnia się kilka odmian technologicznych układów TTL oznaczonych odpowiednio literami:

- **L (ang. Low power)** – wersja o małym poborze mocy (10 razy mniejszym niż typowy TTL), ale wolniejsza od standardowej (10 MHz); nigdy nie zyskała popularności, gdyż została niemal natychmiast zastąpiona układami CMOS serii 4000.
- **H (ang. High speed)** – wersja o większej prędkości od standardowej (58 MHz), ale również o większym poborze mocy. Większą szybkość uzyskano przez zastosowanie 2 razy mniejszych rezystorów, co powodowało przyspieszenie przełączania tranzystorów.
- **S (ang. Schottky)** – odmiana szybka (125 MHz), której tranzystory zawierają dodatkową diodę Schottky’ego włączoną równolegle do złącza kolektor-baza i zabezpieczającą tranzystor przed nasyceniem, co powoduje dużo szybsze przechodzenie tranzystora ze stanu przewodzenia do zatkania.
- **AS (ang. Advanced Schottky)** – ulepszona seria S, charakteryzuje się jeszcze większą szybkością działania.
- **LS (ang. Low power Schottky)** – wersja S o znacznie niższym poborze prądu, zbliżonym do standardowej bramki; główna seria układów TTL używana w większości zastosowań.
- **ALS (ang. Advanced Low power Schottky)** – unowocześniona seria LS z mniejszym poborem mocy.
- **F (ang. Fast)** – nowoczesna, najszybsza seria TTL (125 MHz).

Układy te można łączyć ze specjalnie produkowanymi układami w technologii CMOS, które są zgodne końcówkowo z TTL o takich samych oznaczeniach i wyróżnione literami C, AC, HCT, HC itp., np. 74HC00. Ponadto układy CMOS w wersji HCT mają takie same poziomy stanów logicznych jak TTL, przez co w łatwy sposób można łączyć je ze sobą.