

PKE – Praktyczny Kurs Elektroniki

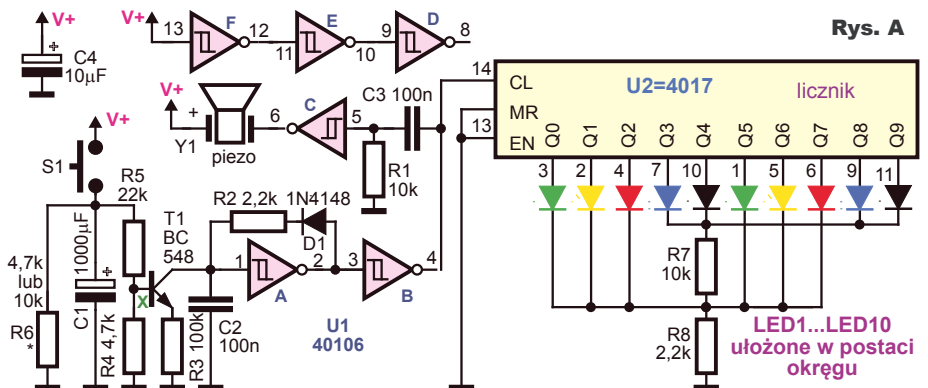


Fotografia tytułowa pokazuje układ prostej ruletki elektronicznej, której schemat zamieszczony jest na **rysunku A**. Naciśnięcie przycisku S1 spowoduje szybkie wirowanie i migotanie punktu świetlnego, czemu towarzyszy charakterystyczny terkot. Po zwolnieniu przycisku prędkość wirowania zmniejsza się, a po kilku sekundach punkt świetlny się zatrzymuje. Częstotliwość terkotania zmniejsza się, ponieważ pojedynczy stuk występuje wtedy, gdy zaświeca się kolejna dioda LED. Działanie układu można zobaczyć na filmie, umieszczonym w Elportalu (www.elportal.pl/pke)

Opis układu dla „zaawansowanych”

Na schemacie z **rysunku A** możemy wyróżnić szereg bloków, znanych nam już z poprzednich wykładów. Mamy tu licznik 4017 (U2), który zlicza impulsy podawane na nóżkę 14 i zaświeca kolejne diody LED, dołączone do wyjść. Diody te powinny być tak ułożone, żeby tworzyły kształt okręgu.

Bramka U1A wraz z elementami D1, R2, T1, R3 tworzy znany z poprzedniego wykładu przetwornik napięcia na częstotliwość. Po naciśnięciu przycisku S kondensator C1 zostaje naładowany, a pełne napięcie zasilające podane jest na dzielnik R5, R4. Napięcie na bazie T1 wynosi około 18% napięcia zasilającego, a tranzystor T1 jest źródłem prądowym. Znaczny prąd tego źródła powoduje, że częstotliwość generatora jest dość duża. Po zwolnieniu przycisku S1 kondensator zaczyna się rozładowywać



wywać przez R5 i R6. Napięcie maleje, maleje też prąd tranzystora T1 i zmniejsza się częstotliwość generatora oraz prędkość wirowania punktu świetlnego. W pewnym momencie tranzystor zostaje zatkany i punkt świetlny się zatrzymuje.

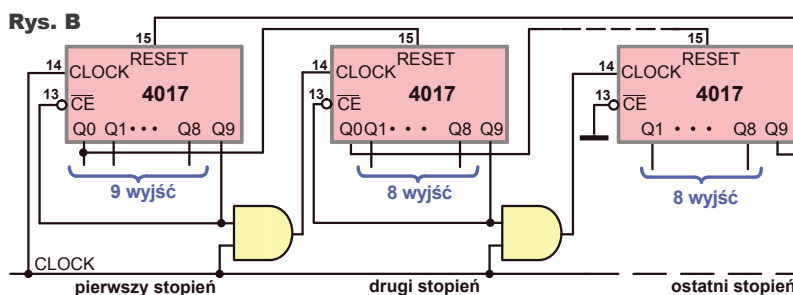
Każde aktywne narastające zbocze na wejściu CL licznika 4017 powoduje nie tylko zaświecenie kolejnej diody, ale też wywołuje krótki impuls dźwiękowy, stuk brzęczyka Y1.

Oczywiście możesz zmieniać wartości elementów. Stała czasowa R1C3 decyduje o głośności i tonie stuku. Częstotliwość głównego generatora, czyli maksymalna szybkość „wirowania” wyznaczona jest głównie przez elementy C2 i R3. Czas trwania cyklu możesz regulować, zmieniając wartość R6.

W tej podstawowej, bardzo prostej wersji ruletki jest 10-pozycyjna. Można dowolnie zwiększyć liczbę diod, zwiększając pojemność licznika. Trzeba wtedy zastosować więcej niż jedną kostkę 4017 i połączyć je w sposób, pokazany w karcie katalogowej licznika 4017 – **rysunek B**.

Aby otrzymać pełnowartościową ruletkę elektroniczną, należałoby jeszcze dodać obwód skutecznego „zatrzymywania”. W przedstawionej najprostszej wersji na końcu cyklu może się zdarzyć, że punkt świetlny dodatkowo przeskoczy o jedną pozycję po pewnym czasie, z uwagi na płynne zatykanie tranzystora T1. Aby temu zapobiec, należałoby dodać obwód zatrzymujący zliczanie, na przykład według **rysunku C**. Wartość rezystora oznaczonego gwiazdką należy dobrać, zależnie od napięcia zasilania, żeby zatrzymanie ruchu punktu świetlnego następowało podobnie jak w klasycznej ruletce.

Na bazie prezentowanego układu można też zrobić inne efekty świetlne. Można na przykład wykorzystać generator przebiegów pseudoprzypadkowych, wykorzystany



wcześniej w wykładzie 22 w elektro- nicznej świeczce. W układzie z **rysunku D** z trzech dotychczas niewykorzysta- nych inwerterów budujemy taki genera- tor, który będzie zmieniał częstotliwość generatora VCO na bramce U1A. Otrzy- mujemy „szalony kręciołek”, w którym częstotliwość wirowania diod zmienia się w dziwny sposób. **Fotografia E** pokazuje model, a w Elportalu można znaleźć filmik, pokazujący jego działanie. Zmiany częstotliwości byłyby skokowe, aby je „złagodzić”, między punkt X i masę został włączony kondensator C8 o pojemności 100uF. Także i tu możesz zmieniać warto- ści elementów RC.

Poznajemy elementy i układy elektroniczne

W poprzednim wykładzie wspomnieliśmy o tym, że dziś realizacja w jednym ukła- dzie scalonym tysięcy, milionów, a nawet miliardów bramek nie jest problemem i że istnieje mnóstwo układów o bardzo skomplikowanej strukturze wewnętrznej, które są swego rodzaju „półproduktami”, ponieważ można je samodzielnie zaprogramować, by pełniły potrzebne, często bardzo skomplikowane zadania.

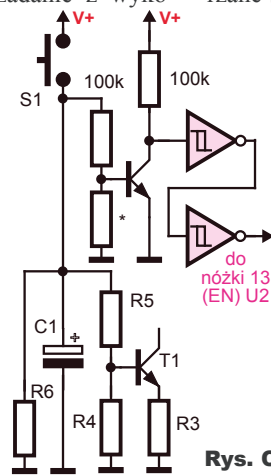
Dowiedziałeś się, że dwie główne grupy to: *układy PLD* oraz dużo popular- niejsze *mikroprocesory*. Programowanie układów PLD polega na takim połączeni- u elementarnych „cegiełek”, by uzyskać strukturę logiczną, która będzie realizo- wać postawione zadanie. Natomiast pro- gramowanie mikroprocesorów nie polega na zmianie ich struktury wewnętrznej, tylko na wpisaniu do ich pamięci pro- gramu, przepisu działania, który będzie realizował postawione zadanie z wyko- rzystaniem dostępnych w danym procesorze struktur.

Do realizacji prost- szych funkcji wystar- czą mikroprocesory 8-bitowe, w których przetwarzane są dane ośmiobitowe. Większe możliwości i więk- szą szybkość realiza- cji zadań zapewniają procesory 16-bitowe, a w komputerach stosowane są procesory 32- i 64-bitowe. O wydajności (mocy

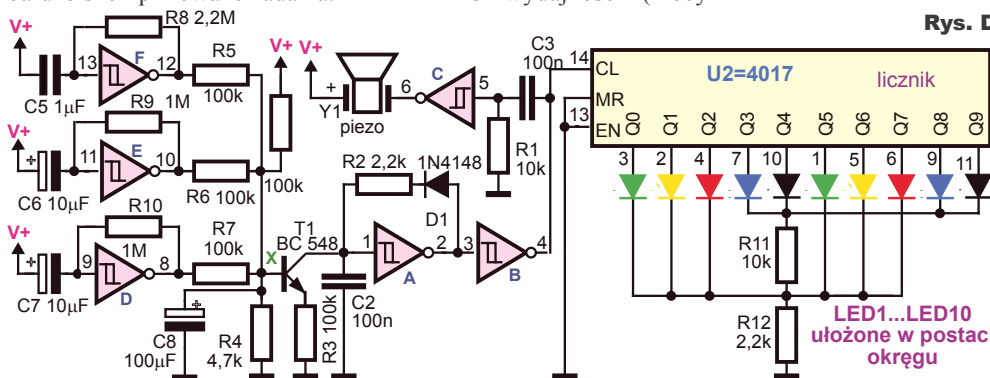
obliczeniowej) decyduje zarówno liczba jednocześnie przetwarzanych bitów, jak też częstotliwość sygnału zegarowego. Wydajne procesory, stosowane w kompu- terach, a także coraz częściej w tabletach i smartfonach, jako główne zadanie mają przeprowadzanie obliczeń matematycz- nych. Można powiedzieć, że takie „duże procesory” to tylko szybka i wydajna „goła maszynka do obliczeń matematycz- nych”. W sposób matematyczny przetwa- rzane są zarówno sygnały obrazu, dźwię- ku, jak też inne.

Natomiast do wykonania wielu innych zadań wystarczą „małe” procesory 8-bitowe, co najwyżej 16-bitowe. Już od dawna, dla wygody konstruktorów, takie prostsze procesory, w jed- nym układzie scalonym, oprócz „gołej maszynki obliczeniowej”, zawierają też dodatkowe bardzo przyteczne bloki pomocnicze, dzięki czemu w jednym układzie scalonym mamy wszystko, co potrzeba do wykonania nawet dość złożonego zadania. Są to **mikroprocesory jednoukładowe**, a ponieważ zawierają dodatkowe bloki, często są nazywane **mikro- kontrolerami jednoukładowymi**. Najpopularniejsze mikrokontrolery jednoukładowe zasilane są napięciem w zakresie 1,8...6V, zwykle 3...5V, a ich ceny wyno- szą od kilku do najwyżej kilku- dziesięciu złotych.

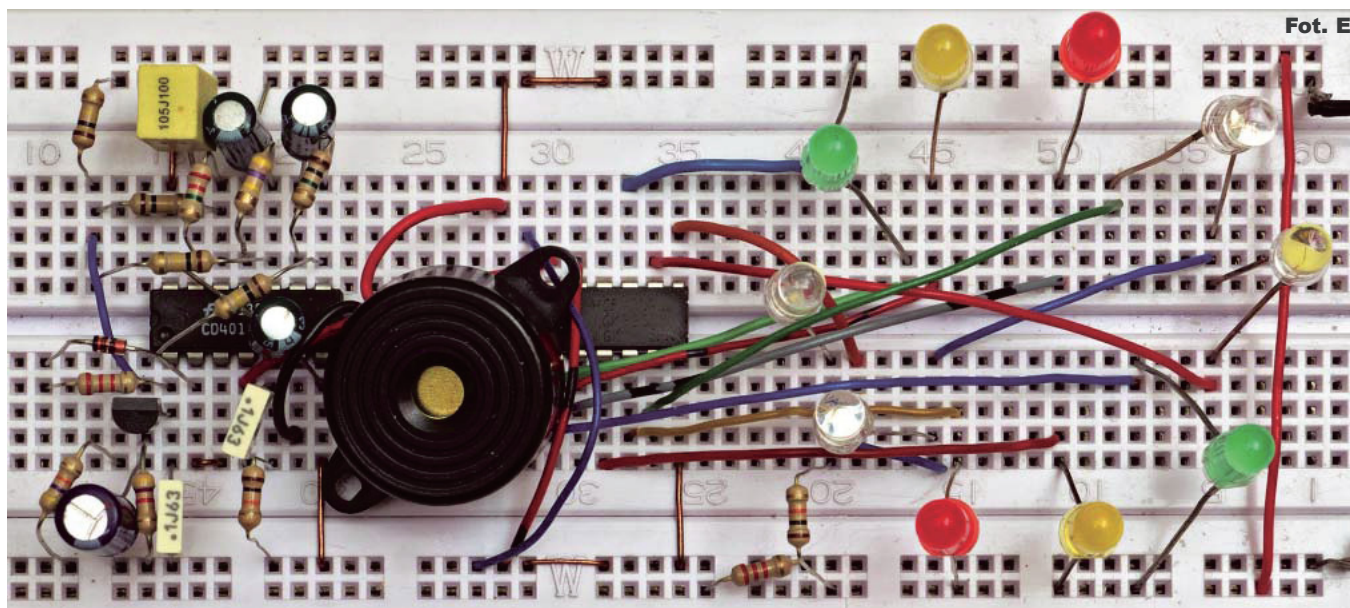
Mikrokontrolery jednoukła- dowe zawierają tzw. *pamięć operacyjną*, potrzebną podczas pracy procesora oraz *pamięć programu*, gdzie trwale przecho- wywany jest „przepis działania”



Rys. C



Rys. D



Fot. E

oraz szereg bloków, które nie są bezpośrednio związane z obliczeniami matematycznymi.

Fotografia 1 pokazuje jednokładowy mikroprocesor ATmega32 popularnej rodziny AVR firmy Atmel w obudowie przewlekanej DIL. **Rysunek 2** pokazuje schemat blokowy tego procesora. Żółta podkładką wyróżniony jest „właściwy procesor”, oznaczony CPU (Central Processing Unit). Zawiera on między innymi jednostkę ALU (Arithmetic Logic Unit), gdzie na liczbach 8-bitowych dokonywane są wszelkie operacje logiczne i matematyczne. Warto dodać, że nazwy *mikrokontroler* i *mikroprocesor* są stosowane wymiennie, choć ściślej biorąc, procesorem należałoby nazywać jedynie jednostkę CPU.

Z CPU współpracują trzy pamięci (na rysunku 2 wyróżnione kolorem czerwonym). Pamięć operacyjna SRAM (Static RAM) to pamięć o bardzo dużej szybkości zapisu/odczytu, ale ulotna – jej zawartość znika po wyłączeniu zasilania. Pamięć ta ma wiele wspólnego ze znanymi nam przerzutnikami, można powiedzieć, że składa się z mnóstwa przerzutników.

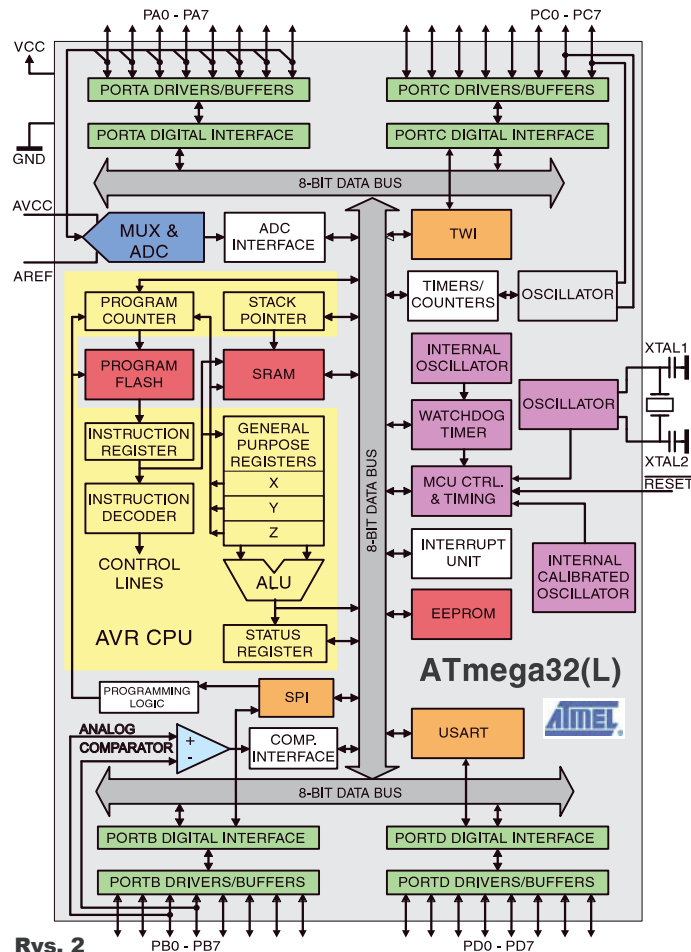
Drugi główny rodzaj to powolniejsza pamięć nieulotna, która zachowuje zawartość po wyłączeniu zasilania przez dziesiątki, a może i setki lat. W mikrokontrolerach są to pamięci FLASH dla programu i nieulotna pamięć pomocnicza EEPROM do trwałego pamiętania niewielkiej ilości danych. Pamięci nieulotne EEPROM, FLASH opierają swoje działanie na dość prostej zasadzie, a mianowicie elementem pamiętającym jest tam... pojedynczy tranzystor MOSFET, gdzie jak wiesz, elektrodą sterującą jest bramka. Napięcie bramki decyduje o tym, czy tranzystor przewodzi, czy jest zatkany. Jednocześnie wiemy, że obwód bramkowy MOSFET-a jest małym kondensatorem, co możemy zobrazować jak na **rysunku 3a**. I właśnie w pamięciach nieulotnych elementami pamiętającymi są w sumie tranzystory MOSFET, ale nietypowe, mające bramkę, ale... bramkę niepodłączoną, zwaną bramką pływającą. Taki tranzystor pamiętający ma jakby wbudowaną „kondensator bramkowy” (C_{GS}), jak poka-



Fot. 1

zuje **rysunek 3b**. Można powiedzieć w uproszczeniu, że jeżeli ten kondensator jest naładowany, to tranzystor pozostaje otwarty. Gdy „kondensator bramkowy” jest rozładowany – tranzystor jest zatkany. W ten sposób mamy możliwość trwałego zapamiętania (na wiele lat) dwóch stanów. Pomimo braku elektrycznego połączenia takiej bramki, można ją „zdalnie” naładować i rozładować. W grę wchodzi tajemnicze zjawisko kwantowe (tunelowe), ale w uproszczeniu można powiedzieć, że zmiana stanu naładowania pojemności bramkowej uzyskuje się za pomocą specjalnych obwodów sterujących i podwyższonych napięć, które jakby „przebijają” cieniutką warstwę izolatora pływającej bramki. Jeden taki tranzystor może zapamiętać (co najmniej) jeden bit informacji, a we współczesnych urządzeniach cyfrowych wykorzystujemy pamięci o pojemności wielu megabajtów, a nawet gigabajtów. Zapewne i Ty masz małego pendrajwa (pendrive), który jest właśnie tego rodzaju pamięcią nieulotną FLASH. Jeżeli przykładowo jego pojemność wynosi „tylko” 1GB (około 1 miliarda bajtów), czyli 8 miliardów bitów, to może on zawierać kilka miliardów takich tranzystorów pamiętających. Natomiast w małych 8-bitowych mikroprocesorach jednokładowych, powszechnie wykorzystywanych przez hobbyistów, nieulotna pamięć programu FLASH zwykle jest wielokrotnie mniejsza i ma pojemność w zakresie 1kB do 256kB. Ulotna pamięć operacyjna SRAM jest jeszcze mniejsza i zwykle ma pojemność od kilkuset bajtów do kilku kilobajtów.

Podczas pracy procesora wykonywane są kolejne rozkazy zawarte w pamięci programu. Odbывается to bardzo szybko – w ciągu sekundy mogą być wykonane miliony prostych, ele-



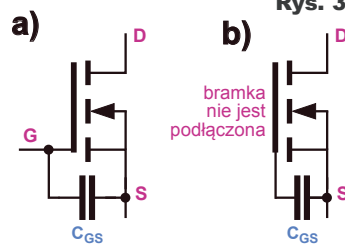
Rys. 2

mentarnych operacji. I właśnie ta ogromna szybkość wykonywania prostych instrukcji umożliwia procesorowi zrealizowanie nawet bardzo skomplikowanych zadań.

Wszystko to odbywa się w takt sygnału zegarowego. Obwody czasowe na rysunku 2 wyróżnione są kolorem fioletowym). Dawniej standardowo generator zegarowy w procesorze współpracował z rezonatorem kwarcowym (XTAL), co zapewniało także dużą dokładność odmierzania czasu. W tym celu przewidziane były/są dwie końcówki do podłączenia rezonatora kwarcowego. Dziś często pracą procesora steruje mniej dokładny wewnętrzny generator zegarowy, którego częstotliwość wyznaczają wbudowane elementy RC.

Mikrokontroler, oprócz „procesora właściwego”, czyli CPU, pamięci i obwodów zegarowych, zawiera też różne inne układy i bloki pomocnicze. Poszczególne bloki komunikują się ze sobą za pomocą wspólnej (8-bitowej) szyny danych (kolor szary).

Do bardzo pożytecznych bloków należą uniwersalne dość skomplikowane liczniki/timery



Rys. 3

bramka nie jest podłączona

(8- lub 16-bitowe), które zwykle mogą zliczać w górę lub w dół i można do nich wpisywać liczby. Dlatego mogą służyć nie tylko jako liczniki zdarzeń czy do odmierzenia czasu, ale także do generowania przebiegu zmiennego o określonym współczynniku wypełnienia, czyli mogą pełnić funkcję generatora – przetwornika PWM.

Standardowym wyposażeniem mikrokontrolera jest też tzw. *watchdog*, który służy do „budzenia” procesora ze stanu uśpienia i może być wykorzystany do przywrócenia prawidłowej pracy po wystąpieniu błędów oraz do zdecydowanego zmniejszenia zużycia energii (procesor „śpi” i jest „budzony” co jakiś czas na krótką chwilę, by zrealizować zadania). Praktycznie każdy mikroprocesor ma wejście, końcówkę zerującą RESET, która pozwala rozpocząć prawidłową pracę procesora od początku programu.

Pozostałe końcówki mikroprocesora mogą pełnić rozmaite funkcje, stosownie do tego, jak zostały skonfigurowane na początku pracy programu lub później. Te pozostałe końcówki zorganizowane są najczęściej w ośmiobitowe porty (kolor zielony), które mają sprytną, dość skomplikowaną budowę, ponieważ zależnie od programu mogą pełnić rozmaite funkcje.

Otóż końcówki portów z zasady mogą pracować jako klasyczne wejścia albo wyjścia (także trójstanowe), na których występują sygnały cyfrowe. Ale niektóre linie portów mogą też pełnić inne, „nietypowe” funkcje. Często mikroprocesor ma współpracować z czujnikami i przetwornikami analogowymi. Wtedy trzeba sprawdzać, a także mierzyć nie tylko obecność czy brak, ale też wartość napięcia. Do sprawdzenia/porównania wartości napięcia może posłużyć *komparator analogowy*, wyróżniony na rysunku 2 kolorem jasnoniebieskim.

Dla ułatwienia tego rodzaju zadań wiele mikrokontrolerów ma też wbudowany wewnętrzny przetwornik analogowo-cyfrowy (ADC = Analog-Digital Converter), który mierzy napięcie i określa wynik w postaci liczby. Często przetwor-

nik ADC jest jeden, ale poprzedza go analogowy multiplexer (MUX), przez co możliwe jest mierzenie napięć w kilku punktach. Na rysunku 2 obwody (ośmikanalowego) przetwornika ADC wyróżnione są kolorem niebieskim.

Niektóre końcówki portów mikrokontrolera mogą też być wejściami czy wyjściami o specyficznym przeznaczeniu – mają specjalizowane obwody i mogą pełnić funkcję portów standardowych łączy, na przykład popularnego RS-232 (UART), TWI, SPI czy aktualnie najpopularniejszego USB. Bardziej rozbudowane mikrokontrolery jednocukłowe mają więcej portów (liczba nóżek może sięgać 100) i mogą mieć dodatkowe obwody dedykowane do konkretnych celów, na przykład do obsługi zaawansowanego wyświetlacza LCD.

Ale warto wiedzieć, że dostępne są też mikrokontrolery o zredukowanej liczbie końcówek, na przykład w rodzinie Atmel AVR są to ATtiny 4/5/9/10 – **fotografia 4**. Ich struktura wewnętrzna jest analogiczna jak w dużym ATmega32, tylko znacznie uproszczona – **rysunek 5**. Mniej jest obwodów pomocniczych i portów, a i pojemności pamięci są znacznie mniejsze (brak też pamięci EEPROM). W niektórych zastosowaniach wystarczy taki skromny mikroprocesor z sześcioma wyprowadzeniami, z których dwa to zasilanie, a cztery to uniwersalne końcówki wejścia/wyjścia o funkcjach zależnych od programu.

Liczba dostępnych rodzin i wersji mikrokontrolerów jednocukłowych różnych producentów może przyprawić o zawrót głowy. Niemniej procesory danej rodziny, pomimo różnic wyposażenia i możliwości, mają taką samą strukturę oraz zasady programowania.

Wbrew pozorom, programowanie okazuje się proste. Aby ułatwić pisanie programów dla mikroprocesorów jednocukłowych, stworzono programy na komputer PC, tzw. kompilatory, a także całe pakiety wspomagające. Wystarczy zainstalować na PC-cie lub laptopie jeden z takich kompilatorów i za jego pomocą napisać program w tzw. postaci źródłowej (najlepiej z komentarzami, opisującymi działanie

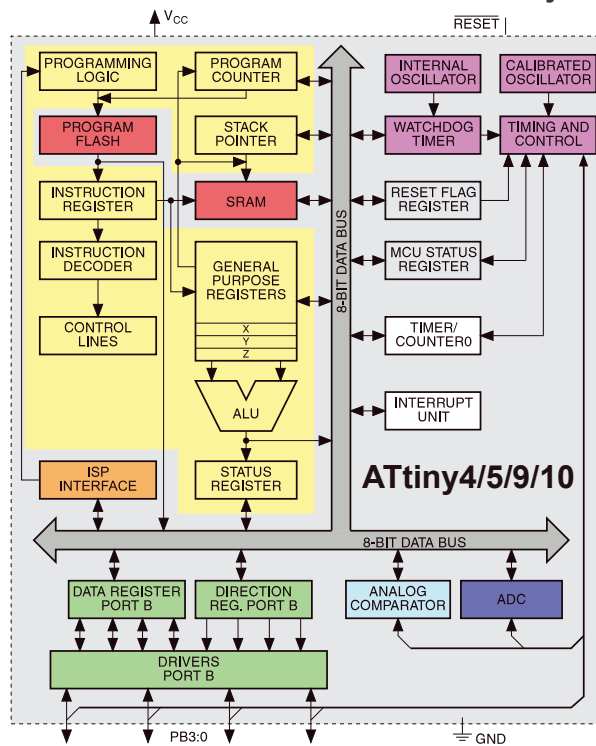
programu). Na tym etapie wykorzystuje się różne języki programowania, w tym popularny i prosty BASIC oraz znacznie lepszy, ale trudniejszy język C. Na **rysunku 6** (z Wikipedii) masz zrzut ekranu z pakietu BASCOM AVR.

Po napisaniu w jednym z języków programowania, program źródłowy należy sprawdzić oraz skompilować, czyli zamienić na postać zrozumiałą dla procesora – na program wynikowy. Ten program wynikowy trzeba wpisać do pamięci FLASH mikrokontrolera.

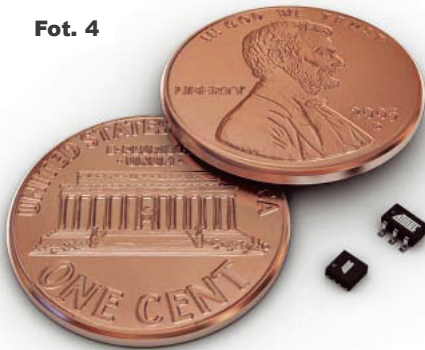
Do zaprogramowania, czyli wpisanie skompilowanego pliku programu do pamięci procesora, najczęściej służy prosta przystawka włączana między port USB komputera PC a końcówką jednego z portów mikrokontrolera. Program mikrokontrolera można zmieniać, modyfikować i wielokrotnie wpisywać do jego pamięci, co jest bardzo użyteczne na etapie tworzenia, ulepszania i usuwania błędów (odpluskwania – *debugging*).

W sumie programowanie procesorów naprawdę nie jest trudne, trzeba jednak zrozumieć działanie procesora, jego budowę oraz poznać podstawy jakiegoś języka programowania i znaczenie poszczególnych rozkazów. Praktyka pokazuje, że z powrodeniem radzą sobie z tym nawet osoby kilkunastoletnie, a także osoby starsze, które w szkole uczyły się elektroniki opartej na lampach elektronowych. Opanowanie programowania mikrokontrolerów daje ogromne możliwości, ograniczone tylko przez wyobraźnię.

Rys. 5



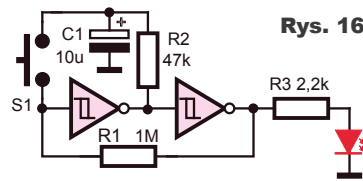
Fot. 4



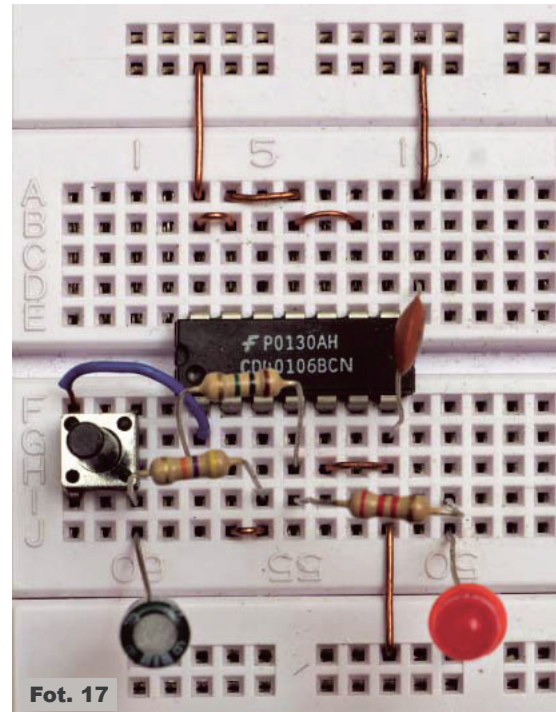
Według rysunku 18 i fotografii 19 możesz zrealizować cyfrowy miernik pojemności. Zawiera on licznik U2 z dziesięcioma diodami LED. Zlicza on impulsy podawane na wejście CL z generatora na bramkach U1D, U1A. Jest zerowany krótkimi impulsami, podawanymi na wejście zerujące MR.

Pracą miernika pojemności steruje generator U1C, R6, C4. Gdy w punkcie X panuje stan niski – potencjał masy, wtedy generator U1D, U1A nie pracuje. Na wyjściu bramki U1B panuje stan wysoki, więc dzięki diodzie D1 w punkcie A panuje napięcie bliskie dodatniemu napięciu zasilania (pomniejszone o około 0,7V – spadek napięcia na D1). Mierzony kondensator Cx jest naładowany. Gdy w punkcie X pojawia się stan wysoki, zaczyna się cykl pracy miernika. Przede wszystkim dodatnie, rosnące zbrocze w punkcie X powoduje wytworzenie przez obwód różniczkujący R3C3 króciutkiego dodatniego impulsu, który zeruje licznik. Jednocześnie stan wysoki w punkcie X i stan wysoki w punkcie A powodują, że zaczyna pracować licznik U1D, U1A. Licznik U2 zlicza jego impulsy.

Na wyjściu bramki U1B występuje wtedy stan niski, a to oznacza, że naładowany wcześniej mierzony kondensator Cx zaczyna się rozładowywać przez

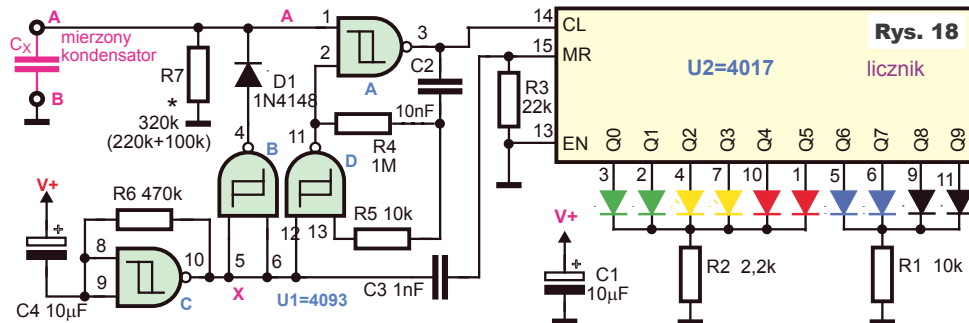


rezystor R7. Napięcie w punkcie A i na nóżce 1 U1A maleje. Licznik U1D, U1A przestanie zliczać, gdy napięcie to zmniejszy się poniżej napięcia progowego bramki U1A. Wtedy generator przestanie pracować i stan licznika zatrzyma się. Gdy znów w punkcie X pojawi się stan niski, sytuacja pozostanie niezmieniona – licznik będzie wciąż pokazywał, ile zliczył impulsów. Dopiero gdy w punkcie X pojawi się stan wysoki, licznik zostanie wyzerowany i cykl pomiarowy zacznie się od nowa. Dzięki temu, że okres pracy generatora U1C wynosi kilka sekund, a generator U1D, U1A ma częstotliwość wielokrotnie większą, podczas pomiaru obserwujemy tylko krótkie mignięcia diod podczas pomiaru. A przez większość czasu miernik wyświetla wynik ostatniego pomiaru. Działanie tego miernika pojemności możesz zaobserwować na filmiku, dostępnym w Elportalu.

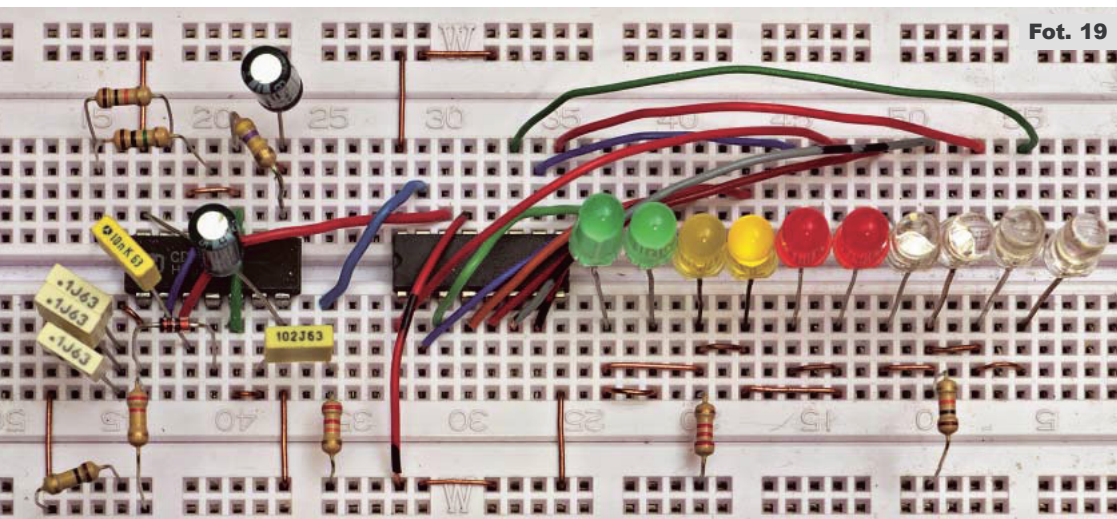


Fot. 17

Dodając do układu komparator analogowy, można byłoby zbudować inne przyrządy. Rysunek 20 pokazuje nieco uproszczony schemat... cyfrowego woltmierz. Licznik U2 zlicza impulsy z generatora, gdy stan wysoki występuje zarówno w punkcie X, jak i w punkcie A. Także i tu pracą układu steruje generator na bramce U1C. Gdy na wyjściu tego generatora, w punkcie Y panuje stan wysoki, a w punkcie X panuje stan niski, wtedy generator U1D, U1A nie pracuje. W punkcie Y panuje stan wysoki, więc prąd płynie przez R7, tranzystor T1 jest nasycony i napięcie na kondensatorze C5, w punkcie Z jest praktycznie równe zero. Układ ten ma zmie-



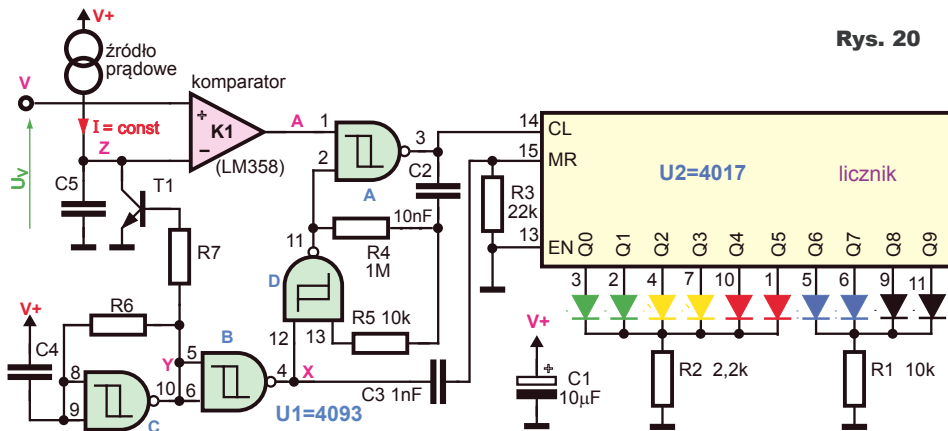
Fot. 19



rzyć napięcie w punkcie V. Załóżmy, że na punkt V podaliśmy z zewnątrz jakieś niewielkie napięcie, np. +1V (oczywiście mierzone względem masy). W punkcie Z napięcie jest równe zero, a w punkcie V +1V, więc na wyjściu komparatora, w punkcie A panuje stan wysoki. Jednak generator nie pracuje, bo w punkcie X panuje stan niski.

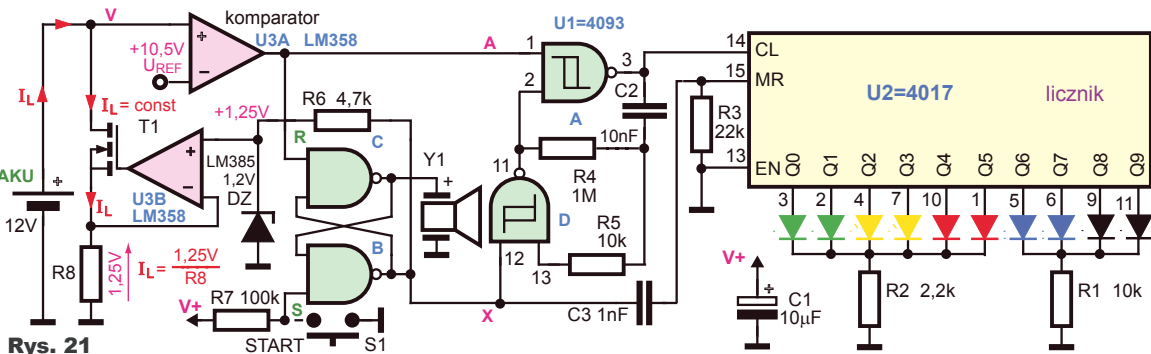
Gdy w punkcie Y pojawi się stan niski, a w punkcie X – wysoki, wtedy zaczyna pracować generator U1D, U1A, obwód R3C3 wytwarza króciutki impuls zerują-

cy licznik U2 i ten zaczyna zliczać jego impulsy, począwszy od zera. Jednocześnie zostaje zatknię tranzystor T1. Kondensator C5 zaczyna się dość szybko ładować niezmiennym prądem źródła prądowego, więc napięcie w punkcie Z rośnie liniowo. Gdy napięcie to przekroczy +1V, czyli stanie się większe niż na wejściu V, wtedy stan komparatora zmienia się: na jego wyjściu, w punkcie A pojawi się stan niski, co zatrzyma licznik U1D, U1A. Licznik zatrzyma się i jedna z diod LED pokaże liczbę zliczonych impulsów. Czym wyższe napięcie mierzone U_V , tym więcej czasu upłynie do zrównania napięć w punktach Z, V



Rys. 20

i licznik zliczy więcej impulsów. Stan licznika jest wprost proporcjonalny do napięcia mierzonego U_V – przy odpowiednim dobraniu pojemności, rezystancji i prądu źródła prądowego układ istotnie będzie (bardzo prostym) woltomierzem.



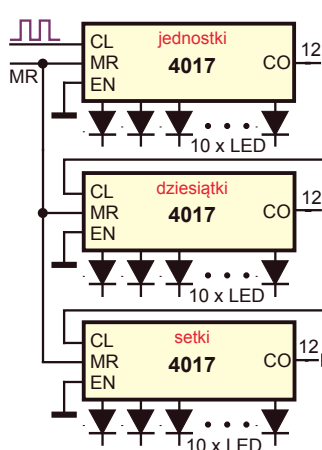
Rys. 21

Inna modyfikacja daje *miernik pojemności akumulatorów*. Pomiar pojemności akumulatora jest czynnością jednorazową: bierzemy w pełni naładowany akumulator, obciążamy go niezmiennym prądem i mierzymy czas, po którym napięcie akumulatora zmniejszy się do wartości granicznej. Pomnożenie tego czasu przez prąd obciążenia daje pojemność akumulatora w amperogodzinach.

Przykładowy miernik pojemności akumulatorów można byłoby zbudować według rysunku 21. Znowu podstawą miernika jest licznik U2 i generator U1D, U1A. Przypuśćmy, że ma to być miernik do kwasowo-ołowiowych akumulatorów 12-woltowych (wtedy napięcie zasilania miernika musiałoby wynosić co najmniej 15V). Do kontroli napięcia akumulatora możemy wykorzystać komparator U3A (np. wzmacniacz operacyjny LM358). Na jego wejście nieodwracające („dodatnie”) podane jest napięcie rozładowywanego akumulatora, a na wejście odwracające („ujemne”) podane jest napięcie odniesienia, równe minimalnemu napięciu bezpiecznego rozładowania. Dla 12-woltowych akumulatorów kwasowych jest to zazwyczaj 10,5V. Jeśli akumulator jest w pełni naładowany, jego napięcie jest wyższe i na wyjściu komparatora, w punkcie A, panuje stan wysoki. Umożliwia to prawidłową pracę generatora U1D, U1A a także właściwe działanie przerzutnika RS z bramkami U1B, U1C.

Ponieważ pomiar jest w tym wypadku czynnością jednorazową, zastosujemy przerzutnik RS, który na początku cyklu ustawimy, naciśkając przycisk START. Miernik zacznie pracę, gdy w punkcie X pojawi się stan wysoki. Znow obwód R3C3 wyzeruje licznik, który zacznie zliczać od zera. Co ważne, dopiero po pojawieniu się stanu wysokiego w punkcie X zacznie płynąć prąd przez rezystor R6 i na diodzie Zenera DZ (raczej na scalonym źródle napięcia odniesienia) wystąpi napięcie dodatnie (+1,25V). Wzmacniacz operacyjny tak występuje tranzystor T1 (MOSFET N), by spadek napięcia na R8 miał taką samą wartość. Wzmacniacz operacyjny U3B z tranzystorem T1 tworzą źródło prądowe – akumulator rozładowywany jest niezmiennym prądem ($I_L = 1,25V/R8$). Wartość rezystora R8 i prądu należy dobrać stosownie do pojemności nominalnej akumulatora, na przykład liczbowo równą 1/5 pojemności nominalnej (wyrażanej w amperogodzinach).

Podczas rozładowania napięcie akumulatora będzie się pomalutku zmniejszać. Gdy spadnie poniżej 10,5V, kom-



Rys. 22

parator U3A zmienia stan wyjścia – pojawi się tam stan niski, co po pierwsze zatrzyma licznik, a po drugie wyzeruje przerzutnik U1B, U1C, przez co w punkcie X pojawi się stan niski. Jednocześnie odezwie się brzęczyk piezo Y1, sygnalizując koniec pomiaru.

Oczywiście takie mierniki byłyby niedokładne. Zamiast pojedynczego licznika U2 należałoby zastosować licznik o większej pojemności, na przykład łącząc kilka kostek 4017 według rysunku 22, by zliczać jednostki, dziesiątki, setki, itd. Należałoby także zadbać o dokładność, stabilizując napięcie zasilania i stosując wysokiej jakości kondensatory i rezystory, żeby zminimalizować problem zmian temperatury i starzenia. Są to obszerne i nietłwne zagadnienia, w które w naszym kursie podstaw elektroniki wgłębiać się nie będziemy.

Kurs PKE dobiegł końca. W 24 wykładach pokazałem Ci zarys współczesnej elektroniki. Jeżeli chciałbyś mi przekazać swoje uwagi czy sugestie, wykorzystaj adres edw@elportal.pl.

Piotr Górecki