

# Ułatw sobie życie dzięki logice, czyli komórki CLC w układach PIC

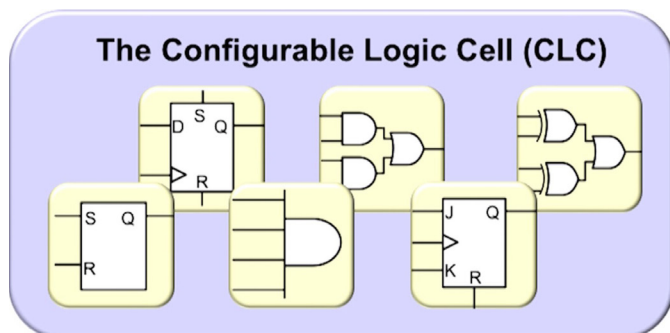
Wiele nowoczesnych systemów wbudowanych działa w oparciu nie tylko o listę złożonych operacji arytmetycznych, zapisanych w postaci kolejnych poleceń, ale także wykorzystuje szereg warunków logicznych. Ich implementacja za pomocą jednostki arytmetyczno-logicznej jest daleka od optymalnej i często nawet proste warunki, jeśli muszą być wielokrotnie sprawdzane, potrafią pożerać niemałą ilość mocy obliczeniowej, a w konsekwencji także energii. Dobrym sposobem na rozwiązanie tego problemu jest wykorzystanie prawdziwych układów logicznych, choć ich umieszczenie na płycie drukowanej bywa kosztowne, a do tego zajmują one dodatkową przestrzeń. Ileż łatwiej byłoby, gdyby taki zestaw układów znalazł się wewnątrz głównego mikrokontrolera urządzenia...

Tu z pomocą przychodzą konfigurowalne komórki logiczne CLC (Configurable Logic Cell). Jakiś czas temu zostały one dodane do zestawu peryferiów mikrokontrolerów PIC firmy Microchip. Pozwalają użytkownikom na projektowanie prostych funkcji, które będą mogły działać w połączeniu z MCU.

## Szybko, łatwo i niezależnie

Praca z komórkami CLC jest prosta. Wystarczy wskazać sygnały, które mają być podawane na wejście, określić funkcje logiczne na tych sygnałach, a następnie zdecydować, gdzie przekazywany ma być wynik operacji. Może on trafić np. na wejście innych peryferiów mikrokontrolera lub bezpośrednio na jego wyjścia. Dzięki temu projektant zyskuje ogromną swobodę konfiguracji wejść i wyjść mikrokontrolera.

Co więcej, aby uprościć sprawę, obsługa komórek CLC została wbudowana w oprogramowanie Microchip MPLAB Code Configurator (MCC). Dzięki temu projektant może konfigurować funkcje logiczne poprzez przeciąganie bramek logicznych i łączenie ich z wejściami i wyjściami za pomocą prostego interfejsu graficznego. Na podstawie wprowadzonych informacji, za pomocą pojedynczego kliknięcia można wygenerować gotowy kod programu w języku C.



Rysunek 1. Komórki logiczne CLC mogą realizować różne funkcje logiczne, a w tym pełnić rolę przerzutników typu D, JK i SR

Komórki CLC to należą do grupy obwodów peryferyjnych, niezależnych od rdzenia (CIP - Core Independent Peripherals). Oznacza to, że pracują bez potrzeby nadzoru ani wykonywania dodatkowego kodu przez rdzeń układu. Upraszczają implementację złożonych systemów sterowania i zapewniają programiście dużą swobodę działania, a jednocześnie redukują obciążenie CPU, zwiększając tym samym sumaryczną wydajność układu. Korzystając z CLC można zaimplementować m.in. mechanizm wykrywania fazy sygnału, generator komplementarny, system monitorowania wielu parametrów w aplikacji, czy też modulator cyfrowy – przykłady te zostały zaprezentowane w dalszej części artykułu. W praktyce, dzięki licznym wejściom, wyjściom i sygnałom wyzwalającym, możliwości wykorzystania komórek CLC są nieograniczone.

## Architektura CLC

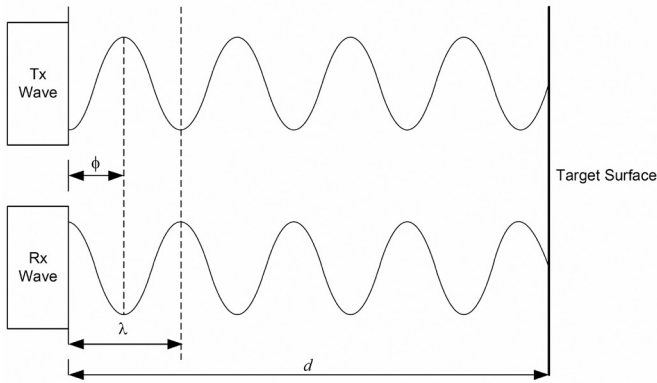
Komórki CLC to obwody konfigurowane przez użytkownika, podobne strukturą do programowalnych układów logicznych (PLD – Programmable Logic Device), ale zintegrowane wewnątrz mikrokontrolera (rysunek 1). Jako wejścia do komórek CLC mogą posłużyć zarówno zewnętrzne wyprowadzenia, jak i sygnały dostępne jedynie wewnątrz obudowy MCU. Przykładowo można przekierować na wejście komórki CLC sygnał z innego obwodu peryferyjnego. Po wykonaniu operacji, sygnał wyjściowy może być przekazany albo na wyprowadzenie układu, albo do innego obwodu peryferyjnego. W efekcie na wejście CLC można podawać np. wewnętrzne sygnały zegarowe lub np. przekierować wyjście timera.

Komórki CLC mogą realizować różne funkcje logiczne, takie jak AND, OR, NOT, XOR, NAND, NOR i XNOR. Sygnał wchodzący do komórki trafia najpierw na blok bramkowania, a następnie jest prowadzony do sekcji odpowiadającej za wybór funkcji logicznych, jakie mają być wykonane. Po przejściu przez funkcje logiczne, sygnał trafia do bloku polaryzacji wyjścia i dopiero wtedy kierowany jest na zewnątrz.

Komórki CLC mogą być wykorzystane jako samodzielne obwody peryferyjne do zaimplementowania sekwencyjnych i kombinacyjnych funkcji logicznych. W efekcie pozwalają na tworzenie szybkich sygnałów wyzwalanych zdarzeniami oraz na błyskawiczne odpowiadanie na zdarzenia. Mogą także wspomagać inne obwody peryferyjne, rozszerzając ich funkcje, stanowiąc sprzętową implementację złożonych operacji logicznych.

Warto dodać, że ponieważ komórki CLC są niezależne od rdzenia, nie tylko zmniejszają obciążenie procesora, ale też pozwalają realizować bardziej zaawansowane funkcje bez zwiększania zapewnienia pamięci Flash i RAM, ponieważ zastępują programowe rozwiązania oparte o algorytmy.

Takie rozwiązanie ma praktycznie same zalety, gdyż funkcje logiczne zaimplementowane sprzętowo są zawsze realizowane szybciej niż takie same funkcje obsługiwane programowo. A fakt, że są w tej samej obudowie co mikrokontroler, pozwala opracowywać rozwiązania nie tylko szybsze, ale i zajmujące mniejszą powierzchnię płytki drukowanej.



Rysunek 2. Pomiar odległości zgodnie z wzorem 1

### Detektor fazy

Uniwersalność i prostota komórek CLC pozwalają rozszerzyć możliwości mikrokontrolerów z rodziny PIC. Dobrym przykładem jest realizacja detektora fazy. Obwód tego typu może być wykorzystany w wielu aplikacjach, jak choćby do pomiarów odległości. Podobne rozwiązania stosuje się też w cyfrowych systemach sterowania mocą, systemach komunikacyjnych i urządzeniach medycznych.

W przypadku pomiarów odległości, wykorzystywana zasada opiera się o fakt, że jeśli ciągła fala radiowa jest transmitowana do celu, odległość do tego celu jest proporcjonalna do przesunięcia fazowego pomiędzy falą nadawaną i odbijaną (rysunek 2). Dzieje się to zgodnie ze wzorem 1:

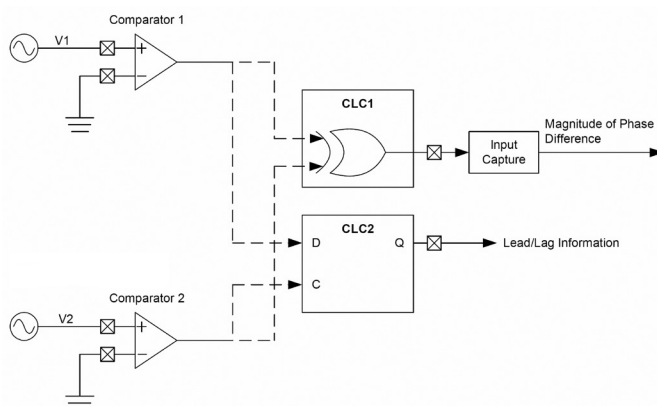
$$d = \frac{\phi\lambda}{4\pi} = \frac{\phi c}{4\pi f}$$

gdzie:

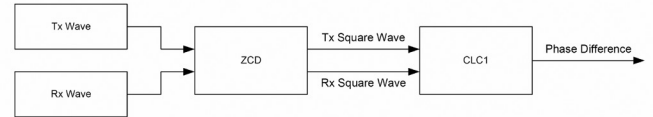
- $d$  – dystans do obiektu,
- $\phi$  – różnica fazy pomiędzy falą nadawaną a odbitą,
- $\lambda$  – długość fali wykorzystanej do pomiaru,
- $f$  – częstotliwość fali wykorzystanej do pomiaru,
- $c$  – prędkość rozchodzenia się fali wykorzystanej do pomiaru,

Komórki CLC mogą zostać użyte do pomiaru różnicy w fazach pomiędzy dwoma sygnałami o tej samej częstotliwości. Na wejścia komórek CLC należy podać sygnał fali transmitowanej i odbieranej, a wynikowa różnica faz, jaka znajdzie się na wyjściu komórki CLC będzie mogła posłużyć do policzenia odległości od źródła sygnału do celu.

Implementacja detektora fazy za pomocą CLC polega na użyciu funkcji logicznych AND-OR do realizacji funkcji XOR, która mierzy bezwzględną różnicę pomiędzy fazami oraz przerzutnika typu D do sprawdzenia, która fala wyprzedza którą (rysunek 3). W algorytmie



Rysunek 3. Układ do detekcji fazy z wykorzystaniem dwóch komórek CLC i opcjonalnie dwóch komparatorów. W przypadku podłączenia komparatora do sieci zasilającej, konieczne jest użycie odpowiednich obwodów zabezpieczających. Linie przerywane na ilustracji reprezentują wewnętrzne połączenia



Rysunek 4. Na potrzeby pomiaru odległości wystarczy zastosowanie prostszego układu

tym nie muszą być konieczne stosowane fale prostokątne. Inne fale, np. sinusoidalne również pozwolą uzyskać dobry wynik.

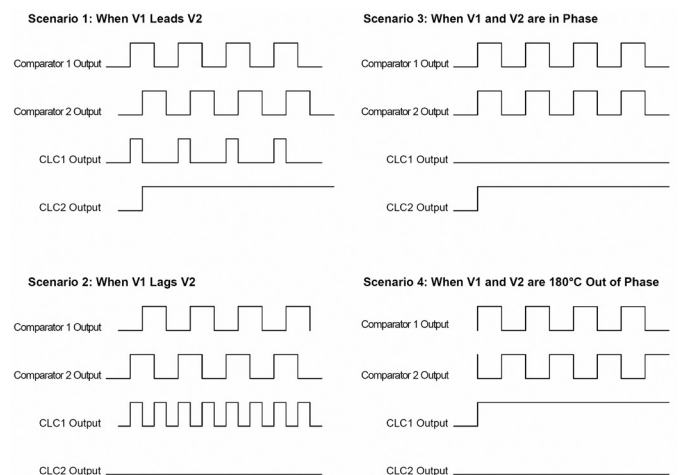
Sygnały źródłowe, których różnica fazy ma zostać zmierzona są podawane na wejścia dwóch komparatorów, które są skonfigurowane do pracy jako detektory przejścia przez zero (rysunek 4). To właśnie one zamieniają wejściowy sygnał analogowy na sygnał prostokątny o tej samej częstotliwości. Jeśli jednak mamy pewność, że sygnał wejściowy jest już sam w sobie prostokątny, wtedy zastosowanie komparatorów nie jest potrzebne. Gotowe sygnały prostokątne są następnie wewnętrznie przekierowywane na wejścia odpowiednich komórek CLC.

W pierwszej z komórek (CLC1) realizowana jest funkcja XOR, której efektem na wyjściu są impulsy o długości proporcjonalnej do bezwzględnej wartości przesunięcia fazowego. Sygnał ten jest kierowany na wejście mikrokontrolera, który mierzy czas jego trwania i w ten sposób oblicza wynik. Jeśli na wyjściu CLC1 nie pojawiają się żadne impulsy, oznacza to że sygnały wejściowe są zsynchronizowane w fazie.

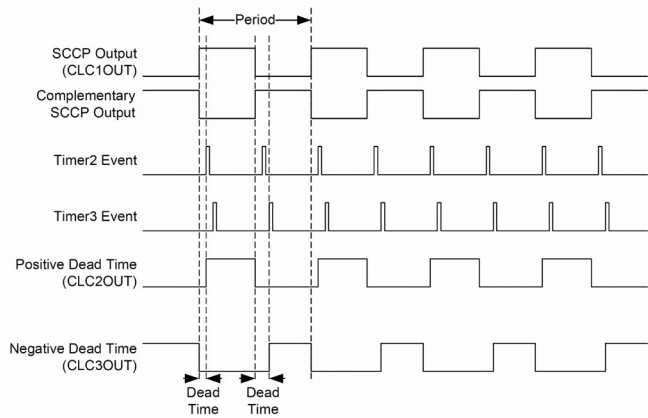
Uzyskana z CLC1 wartość nie wskazuje, która z fal wyprzedza którą. Dlatego konieczne jest użycie drugiej komórki (CLC2), skonfigurowanej do pracy jako przerzutnik typu D, w którym jedna fala podawana jest na jego wejście, a druga jako sygnał zegarowy. Wyjście CLC2, zależnie od swojego stanu, będzie wskazywało, czy sygnał podawany na wejście przerzutnika wyprzedza czy podąża za sygnałem zegarowym (rysunek 5).

### Generator przebiegów komplementarnych

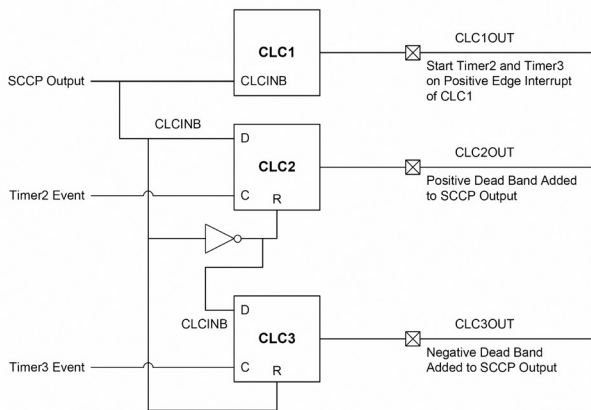
Innym sposobem na wykorzystanie komórek CLC jest stworzenie generatora przebiegów komplementarnych (CWG – Complementary Waveform Generator) z czasem martwym. Czas martwy (dead-band) jest wstawiany pomiędzy dwa sygnały, by zapobiec nadmiernemu wzrostowi prądu w momencie przełączania sygnałów, co ma znaczenie szczególnie w aplikacjach związanych z zasilaniem (rysunek 6). Aplikacja ta stanowi również przykład użycia funkcji wykrywania zbocza za pomocą komórek CLC oraz wprowadzania przerw w generowane przebiegi za pomocą pojedynczego modułu zdejmowania/porównywania/PWM (SCCP - Single Capture/Compare/PWM module) na wejściu CLC (rysunek 6).



Rysunek 5. Przebiegi sygnałów wewnątrz mikrokontrolera w zależności od scenariusza: 1) gdy fala V1 wyprzedza falę V2; 2) gdy fala V2 wyprzedza falę V1; 3) gdy fale V1 i V2 są zsynchronizowane w fazie; 4) gdy fale V1 i V2 są przesunięte w fazie o równo 180°



Rysunek 6. Poszczególne przebiegi w układzie podczas pracy generatora



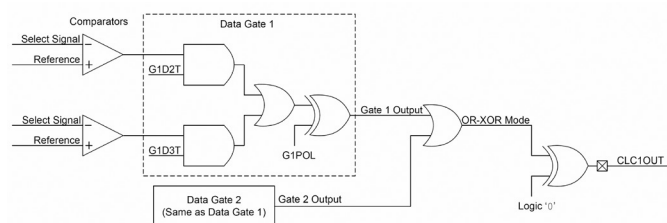
Rysunek 7. Schemat połączeń wewnętrznych dla generatora przebiegów komplementarnych z czasem martwym, pracującym w trybie synchronizacji zbocza; bramka NOR na schemacie jest zrealizowana z użyciem bitów rejestru komórek CLC

Często aplikacje takie jak sterowanie silnikami, wymagają użycia kilku generatorów tego typu do kontroli ich pracy. Moduł MCCP (Multiple Capture/Compare/PWM) pozwala na generację przebiegów komplementarnych, które na siebie nie będą nachodzić, co jest możliwe dzięki ścisłej kontroli długości czasu martwego na wyjściu. Jednakże, jeśli dana aplikacja wymaga większej liczby modułów MCCP niż te dostępne w układzie, wtedy alternatywą jest zastosowanie SCCP w połączeniu z CLC.

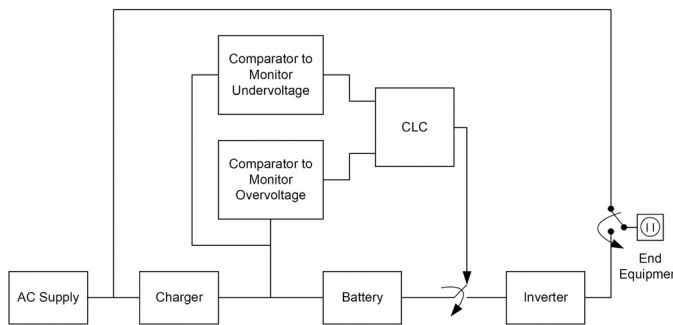
Czas martwy można dodać do wyjść SCCP zarówno, gdy są one ze sobą zsynchronizowane zboczami, jak i środkiem impulsów. Zostało to pokazane na rysunku 7, gdzie trzy komórki CLC sterują długością czasu martwego w aplikacji w trybie synchronizacji zbocza.

**Monitorowanie wielu parametrów jednocześnie**

Często aplikacje wymagają monitorowania wielu różnych parametrów jednocześnie, takich jak np. temperatura, ciśnienie czy wilgotność. Jeśli tylko zaczną one przekraczać określone dolne lub górne granice, konieczne jest podjęcie stosownych kroków.



Rysunek 8. Monitorowanie wielu parametrów z użyciem komórek CLC



Rysunek 9. Monitorowanie napięcia w bezprzerwowym systemie zasilania na wypadek przekroczenia wartości granicznych w górę lub w dół.

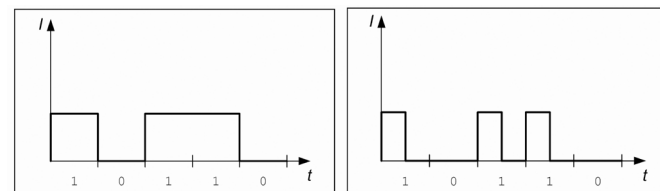
Na wejście komórek CLC można podłączyć sygnały pochodzące z wielu komparatorów. W normalnej sytuacji, w mikrokontrolerze komparator może zostać użyty do monitorowania tylko jednego parametru. Jeśli jednak użyć komórek CLC, wiele wyjść komparatorów może posłużyć do monitorowania różnych parametrów (rysunek 8).

W zależności od zastosowanej konfiguracji, akcje mogą być podejmowane, albo gdy dowolny z parametrów przekroczy zadaną granicę, albo gdy granice zostaną przekroczone na wszystkich z nich. W praktyce układy tego typu mają zastosowanie np. podczas monitorowania temperatury i ciśnienia w zakładzie przemysłowym, albo na potrzeby monitorowania poziomów napięć w systemach zasilania bezprzerwowego (rysunek 9).

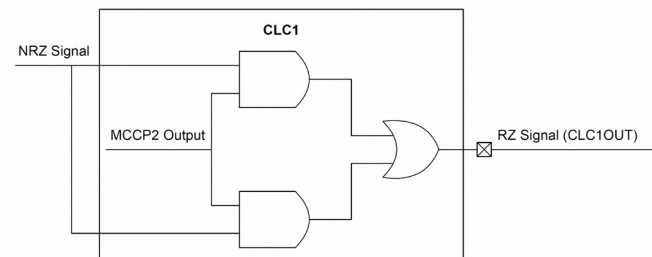
**Kodowanie sygnału cyfrowego**

Komórki CLC można z łatwością wykorzystać do zmiany kodowania sygnału cyfrowego, np. z postaci NRZ (Non-Return-to-Zero - bez powrotu do zera) do postaci RZ (Return-to-Zero). Przykład sygnału przed i po konwersji pokazano na rysunku 10.

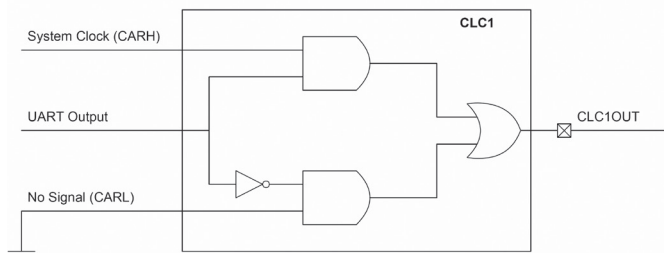
By przekodować sygnał NRZ do RZ potrzebne jest użycie jednej komórki CLC i jednego modułu MCCP (rysunek 11). Przyjmując, że  $T$  to czas trwania jednego bitu w kodowaniu NRZ, moduł MCCP2 należy skonfigurować tak, by przez połowę tego czasu generował wysoki, a przez drugą połowę niski sygnał. Powinien zostać ustawiony do pracy w trybie wyzwalania tak, by wyjście CLC1 było źródłem wyzwalania. Natomiast na wejście CLC1 należy podać sygnał zakodowany NRZ i wyjście modułu MCCP. Komórka CLC1 powinna zostać skonfigurowana do pracy w trybie AND-OR. Wtedy to wyjście MCCP2 będzie stanowiło sygnał zegarowy odniesienia, a na wyjściu CLC1 pojawi się sygnał zakodowany w postaci RZ.



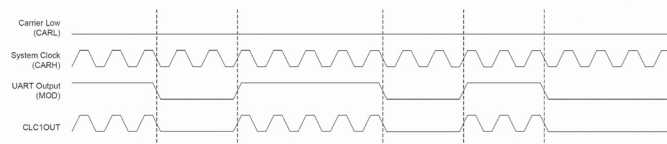
Rysunek 10. Sygnał zakodowany w postaci a) NRZ; b) RZ



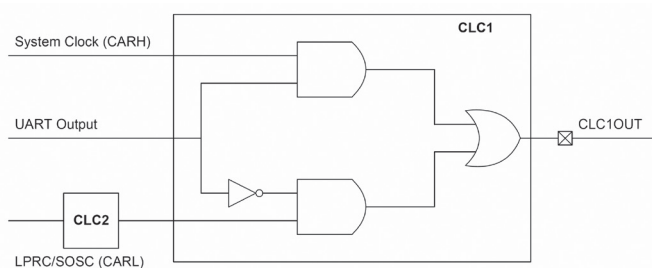
Rysunek 11. Wykorzystanie CLC do przekodowania sygnału NRZ na RZ



**Rysunek 12. Modulator OOK wykonany za pomocą pojedynczej komórki CLC**



**Rysunek 13. Przebiegi uzyskane za pomocą modulatora OOK**



**Rysunek 14. Modulator FSK wykonany za pomocą dwóch komórek CLC**

### Cyfrowa modulacja

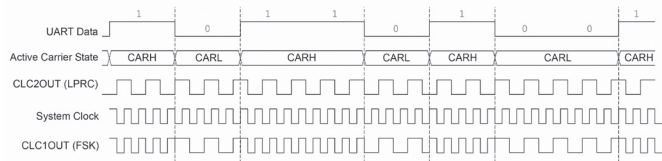
Komórki CLC mogą z łatwością zostać wykorzystane do cyfrowej modulacji sygnału, takiej jak kluczkowanie on/off (OOK – ON-OFF Keying), kluczkowanie częstotliwości (FSK – Frequency Shift Keying) czy kluczkowanie fazy (PSK – Phase-Shift Keying). Do realizacji tych modulacji wystarczy jedna komórka CLC, za wyjątkiem FSK, gdzie potrzebna jest też druga komórka.

W podanym przykładzie, za źródło sygnału modulującego, a więc danych, posłuży sygnał UART, a za nośną – sygnał zegarowy. Oczywiście można modulować inne sygnały innymi sygnałami, bez konieczności wykonywania dodatkowych zewnętrznych połączeń. Wszystkie sygnały można przekierować już wewnątrz układu.

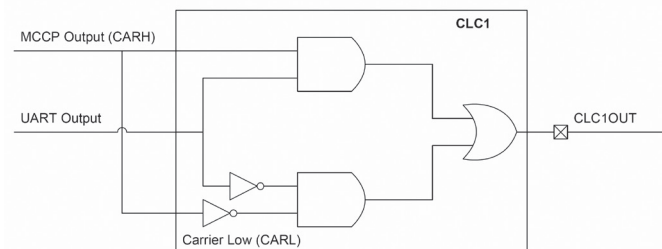
W każdym wymienionym przykładzie modulacji, komórka CLC1 będzie pracować w trybie AND-OR, a na jej wyjściu będziemy uzyskiwać zmodulowany sygnał.

Podczas modulacji OOK na jedno z wejść komórki podajemy sygnał zegarowy, na drugie wyjście UART, a trzecie uziemiemy. Sygnał UART trafia bezpośrednio na wejście jednej z bramek AND oraz po odwróceniu (uzyskanemu poprzez ustawienia w rejestrach komórki) na wejście drugiej z bramek AND. Wewnętrzny schemat pokazano na **rysunku 12**, a na **rysunku 13** widać uzyskane przebiegi.

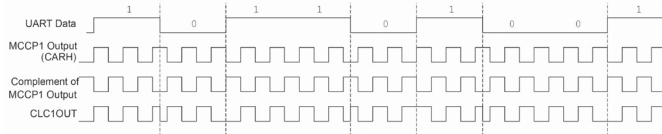
Ponieważ schemat z rys. 12 sprowadza się do tego, że w zależności od aktualnego stanu sygnału UART, na wyjście komórki CLC1 trafia albo sygnał zegarowy albo drugi z podanych, zamiana modulacji OOK na FSK wymaga jedynie podmiany masy z rysunku 12 na sygnał o innej częstotliwości niż zegarowy. Do jego utworzenia można wykorzystać drugą komórkę – CLC2 (**rysunek 14**), która będzie generowała



**Rysunek 15. Przebiegi uzyskane za pomocą modulatora FSK**



**Rysunek 16. Modulator PSK wykonany za pomocą pojedynczej komórki CLC**



**Rysunek 17. Przebiegi uzyskane za pomocą modulatora PSK**

sygnał o dwa razy większym okresie niż okres sygnału zegarowego. Wtedy uzyskamy przebiegi, jak na **rysunku 15**.

W analogiczny sposób uzyskujemy modulację PSK – sygnał prostokątny przesunięty w fazie o 180° można uzyskać odwracając sygnał oryginalny. Ponieważ odwrócenie można zrealizować za pomocą ustawień w rejestrach komórki CLC1, komórka CLC2 przestaje być potrzebna. Schemat modulatora PSK przedstawiono na **rysunku 16**, a uzyskane przebiegi na **rysunku 17**. W tym wypadku, zamiast sygnału zegarowego użyto sygnału wygenerowanego w bloku M CCP.

### Podsumowanie

Dodanie konfigurowalnych komórek logicznych do zestawu obwodów peryferyjnych mikrokontrolerów PIC znacząco zwiększyło ich możliwości. Nie tylko pozwoliło na szybką realizację operacji logicznych, bez potrzeby obciążania czy nawet budzenia mikrokontrolera, ale też zdecydowanie rozbudowało potencjał innych peryferiów. Bardzo korzystne jest również, że takie układy pozwalają ograniczyć przestrzeń zajmowaną przez podzespoły na płycie drukowanej. Wśród obecnie produkowanych mikrokontrolerów PIC dostępne są modele z 1, 2, 3, 4, lub 8 konfigurowalnymi komórkami logicznymi, a więc w ramach jednego układu da się zaimplementować nawet znacznie bardziej skomplikowane obwody logiczne, niż zaprezentowane w niniejszym artykule. I co więcej – fakt, że konfiguracja komórek CLC odbywa się z poziomu wygodnego GUI, w ramach oprogramowania MPLAB Code Configurator (MCC) tym bardziej ułatwia korzystanie z nich.

**Marcin Karbowniczek, EP**

*W artykule wykorzystano niektóre materiały z noty aplikacyjnej autorstwa Manu Venkatesh Gowdy – inżyniera aplikacyjnego z firmy Microchip Technology Inc.*

**[www.ep.com.pl/kap](http://www.ep.com.pl/kap)**