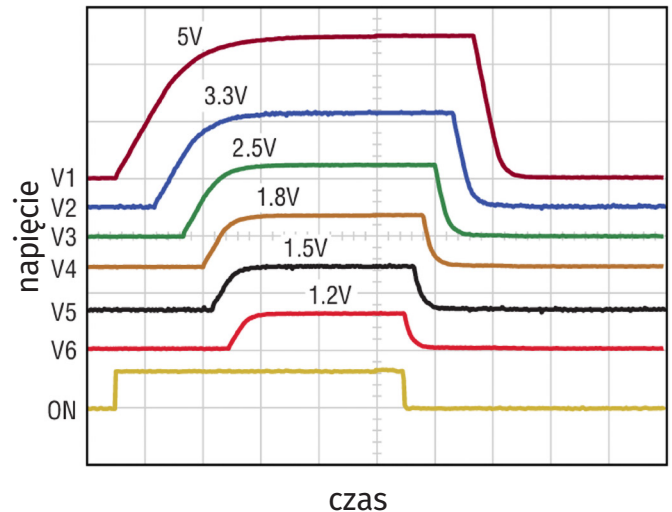


# MOCna strona FPGA

## czyli odpowiedzi na pytania „ile?” i „jak?” przy zasilaniu układów programowalnych (2)

Zasilacze układów FPGA należą do najbardziej złożonych systemów elektronicznych, jakie stosuje się do dostarczania prądu do urządzeń półprzewodnikowych. Muszą być bardzo precyzyjne, szybkie i łatwe do kontrolowania. Jednocześnie oczekujemy, że sekcje zasilania układów programowalnych będą coraz mniejsze, bardziej energooszczędne i będą zawierały coraz bardziej zaawansowane algorytmy sterowania i programowej kontroli dostarczanych napięć. Sprawę komplikuje dodatkowo fakt podziału FPGA na wiele bloków składowych, z których każdy zasilany jest osobnym, innym napięciem. W drugiej części cyklu omówimy układy do załączania tych napięć w odpowiedniej kolejności.



Poradnik opisuje, na przykładzie układów Arria 10, produkowanych przez Alterę, w jaki sposób należy projektować systemy zasilania dla układów półprzewodnikowych tego typu. W ramach artykułu omówione zostaną zasady zasilania układów oraz optymalne rozwiązania systemów zasilania, oparte na scalonych kontrolerach zasilania, takich jak np. LTC3887, LTC2977 i LTM4677 dostarczane przez firmę Analog Devices.

W pierwszej części poradnika pisaliśmy o statycznych parametrach zasilania, szczególnie o wartości napięcia zasilającego i wymaganej dokładności stabilizacji tego napięcia. Opisany algorytm doboru napięcia zasilającego pozwala nie tylko na uzyskanie optymalnej wydajności obliczeniowej układu FPGA, ale także zapewnienie możliwie energooszczędnej pracy.

Redukowanie wymiarów tranzystorów pozwala na konstrukcję szybszych bloków logicznych, jednak elementy te są o wiele delikatniejsze. Współczesne układy programowalne wymuszają wiele kompromisów w tym zakresie, głównie dzieląc cały układ na niezależnie zasilane bloki o różnych funkcjach. Każdy z tych bloków ma swoją, niezależną specyfikację napięcia zasilania. Bloki te – tzw. domeny zasilania – muszą być załączane w ściśle określonej kolejności, inaczej układ może ulec uszkodzeniu. W tej części przyjrzymy się sekwencjonowaniu zasilania, czyli temu, w jaki sposób zadbać, aby kolejność zasilania poszczególnych domen FPGA była zgodna z wymaganiami przedstawionymi w karcie katalogowej układu.

### Poradnik sekwencjonowania zasilania

Niektóre spośród najnowocześniejszych układów FPGA mogą wymagać nawet kilkunastu różnych linii zasilających, które różnią się specyfikacjami prądu, tętnień czy maksymalnego dopuszczalnego szumu. Oprócz tego specyfikacja domen zasilania obejmuje także kolejność sekwencji podawania napięć podczas rozruchu układu, jego wyłączenia i różnych stanów awaryjnych.

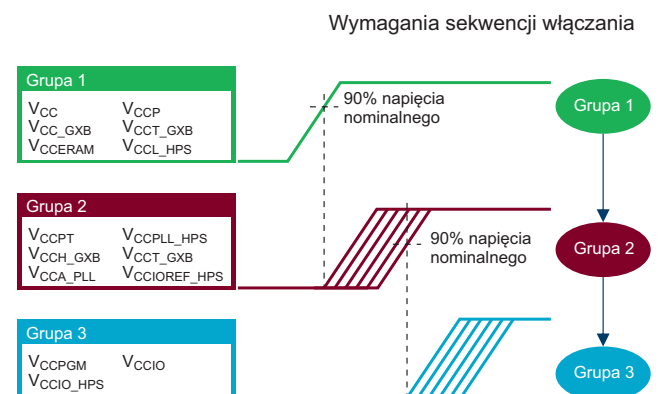
Najnowsze specyfikacje FPGA określają szczegółowe wymagania dotyczące kolejności sekwencji podczas uruchamiania i wyłączenia zasilaczy. Wszyscy wiodący producenci tych elementów, tacy jak Xilinx czy Altera, zalecają określony porządek zasilania

i synchronizację poszczególnych linii, aby zapewnić prawidłowe zresetowanie układu FPGA, utrzymanie minimalnego poboru prądu i utrzymanie odpowiedniej konfiguracji trójstanowej linii wejścia/wyjścia podczas uruchamiania czy wyłączenia zasilania. Biorąc pod uwagę znaczącą liczbę zasilaczy wymaganych przez układ FPGA, złożoność zadania sekwencjonowania jest znaczna.

W przypadku układów Altera Arria 10 specyfikacja dzieli zasilacze na trzy grupy sekwencji (opisanych na **rysunku 1**, jako 1, 2 i 3) i wymaga, aby były one załączane kolejno (1, 2 i 3) podczas włączania i w odwrotnej kolejności odłączane (3, 2 i 1) podczas wyłączenia FPGA.

Podobne zalecenia dotyczą układów od Xilinksa. Sekwencja rozruchowa FPGA Virtex UltraScale sugeruje odpowiednią kolejność napięć:  $V_{CCINT}/V_{CCINT\_IO}$ ,  $V_{CCBRAM}$ ,  $V_{CCAUX}/V_{CCAUX\_IO}$  i finalnie  $V_{CCO}$ . Sekwencjonowanie podczas wyłączenia układu jest odwrotnością podanej sekwencji. Więcej informacji znaleźć można w dokumentacji, dostarczanej przez firmę Xilinx.

Opisane powyżej przypadki to tylko dwa z wielu dostępnych układów FPGA. Prawie każdy nowoczesny system FPGA ma wiele szyn zasilających, a jednym z najbardziej oczywistych pytań jest – w jakiej



Rysunek 1. Uproszczony schemat sekwencjonowania zasilania dla układów z rodziny Altera Arria 10

kolejności powinny się włączać i wyłączać? Nawet jeśli nie ma wyraźnego wymogu sekwencjonowania, istnieją dobre powody, by wymusić deterministyczną sekwencję uruchamiania poszczególnych zasilaczy w systemie. Istnieje szereg sposobów, aby zrealizować sekwencyjne załączanie linii zasilania. Oto niektóre z dostępnych opcji projektowych:

- **Bez sekwencjonowania** – zasilacze uruchamiają się same w swoim tempie, a napięcia podnoszą się i opadają zgodnie ze swoim rytmem. Co mogłoby pójść nie tak?;
- **Sprzętowe sekwencjonowanie kaskadowe** – każdy uruchamiany zasilacz jest podłączony do wejścia załączającego kolejny, aby go uruchomić. Działa to dobrze, ale tylko wtedy, gdy napięcia ułożone są w kolejności rosnącej;
- **Seqwencjonowanie oparte na CPLD** – używa programowalnej logiki, aby stworzyć niestandardowe rozwiązanie. Jest to bardzo elastyczne podejście, ale niesie ze sobą spore wyzwanie dla projektanta sekcji zasilania;
- **Seqwencjonowanie oparte na zdarzeniach** – rozwiązanie jest podobne do sekwencjonowania kaskadowego, ale bardziej elastyczne, ponieważ może działać zarówno przy napięciach malejących, jak i rosnących. Układ scalony sekwencera może być również programowalny i zajmuje się wieloma scenariuszami, jak np. awaria lub skrajne przypadki pracy systemu;
- **Seqwencjonowanie oparte na czasie** – sekwencer zależy od czasu uruchamiania każde kolejne napięcie po odczekaniu określonego czasu od załączenia się poprzedniej linii zasilania. W połączeniu z kompleksowym zarządzaniem sytuacjami awaryjnymi, tego rodzaju układy mogą być bardzo elastyczne, a jednocześnie deterministyczne i bezpieczne.

**Brak sekwencjonowania**

Możliwe jest włączenie systemu zasilania bez żadnego zarządzania załączaniem poszczególnych domen. Gdy główne zasilanie zostaje podłączone do układu, stabilizatory zaczynają swoją pracę. Analogicznie, po odłączeniu zasilania lub wyłączeniu włącznika przestają one działać i napięcie zanika. Problemów z takim podejściem jest wiele, a niektóre nie są wcale oczywiste.

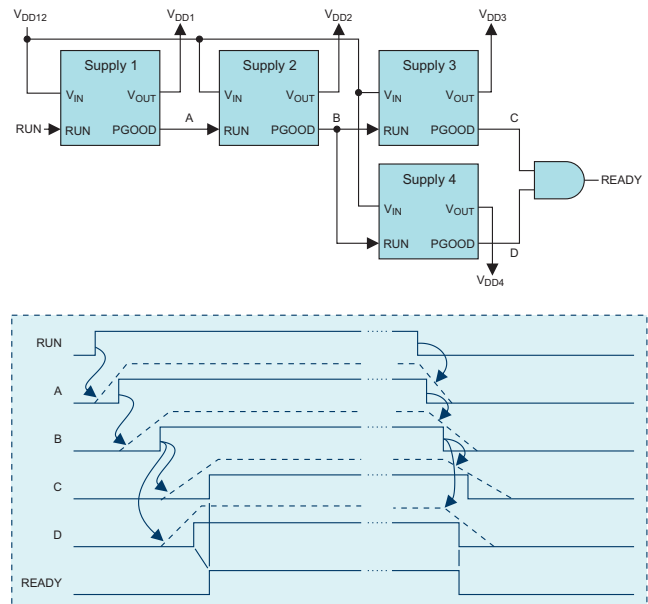
Brak determinizmu czasowego może mieć różne skutki dla systemu. Pierwszym jest nadmierne nadwyrężanie struktury wrażliwego jednak układu FPGA. Może to spowodować natychmiastową katastrofalną awarię albo po prostu przedwczesne starzenie układu. Dodatkowo brak sekwencjonowania zasilania może również powodować nieprzewidziane zachowanie się układu przy resecie po włączeniu lub przyjmowanie nieokreślonych stanów logicznych przez wejścia i wyjścia. Stabilność takiego systemu jest wątpliwa, a powstające problemy trudne do debugowania.

Podejście to sprawia, że pytania dotyczące wykrywania i reagowania na błędy, zarządzania energią czy wsparcia debugowania układu pozostają całkowicie bez odpowiedzi. Unikanie sekwencjonowania zasilania, szczególnie w rozbudowanym układzie FPGA, zawsze niesie ze sobą poważne problemy.

**Seqwencje kaskadowe**

Nieco bardziej uporządkowanym podejściem do sekwencjonowania napięcia zasilania jest klasyczna kaskada powstająca przez podłączenie wyjść PGOOD do wejść RUN, pokazana na **rysunku 2**. Przypomina kostki domina: każda z nich przewraca następną w sekwencji, co gwarantuje załączanie domen zasilania w odpowiedniej kolejności. Zaletą tej techniki jest prostota budowy układu. Niestety ma to też swoje wady. Chociaż często działa dostatecznie dobrze przy sekwencjonowaniu załączania zasilania, to nie może działać w odwrotnej kolejności (ani w żadnej innej kolejności niż ustalona sprzętowo). Może być tylko jedna kolejność sekwencji.

Ponadto taka topologia nie może w prosty sposób obsługiwać sytuacji awaryjnych ani zarządzać energią w innych niż normalne



**Rysunek 2. Seqwencja kaskadowego załączania zasilania**

warunkach pracy. Układ nie jest wystarczająco mądry, aby podejmować jakiegokolwiek decyzje. Jeśli jeden etap podczas sekwencji nie powiedzie się, co będzie dalej? Co się stanie, jeśli jeden z zasilaczy się zepsuje? Odpowiedzi na te pytania pozostają niezdefiniowane, a debugowanie tego rodzaju problemów nie jest łatwe.

Z uwagi na powyższe problemy zasilacze z kaskadową sekwencją stosuje się tylko w najprostszycy urządzeniach z niewieloma domenami zasilania, gdzie ewentualne debugowanie systemu nie jest złożone, a dodanie do sekcji zasilacza prostej logiki, np. w celu łączenia obsługi awaryjnych sytuacji czy zarządzania oszczędzaniem prądu, jest możliwe bez nadmiernego komplikowania układu.

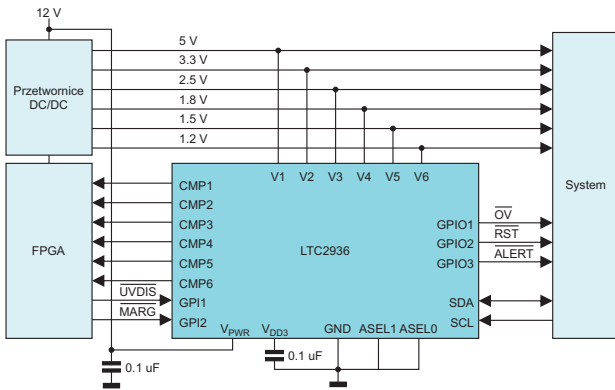
**Zastosowanie FPGA lub CPLD do sekwencjonowania**

Jeżeli chcemy skonstruować kaskadowy sekwencer do zasilania, który obsługuje bardziej zaawansowane zachowanie, dodajemy do niego pewną logikę sterującą. Jeżeli rozrasta się ona i staje się coraz bardziej skomplikowana, możemy dyskretne elementy logiczne zastąpić np. modułem CPLD lub prostym FPGA (na pewno nie takim, który sam wymaga złożonej sekwencji załączania zasilania).

Używanie pomocniczego CPLD lub FPGA w zasilaczu do sekwencjonowania domen zasilania jest opcją wybieraną przez wielu projektantów. W systemie zaprojektowanym przez i dla projektantów układów cyfrowych ma to pewien urok. Naturalnym rozwiązaniem jest zaprojektowanie cyfrowego bloku sterującego w układzie programowalnym, który steruje systemem zasilania innego układu FPGA. Oczywiście, nie jest to tak proste, jak może wydawać się na początku, ponieważ systemy zasilania należy rozpatrywać z szerszej perspektywy niż tylko samo sterowanie cyfrowe.

Jeśli projektanci chcą rozwiązać problem sekwencjonowania i zarządzania zasilaniem, muszą najpierw dokładnie zrozumieć jego złożoność. Omówiliśmy już wiele z tych aspektów, ale jest ich więcej. Są to zagadnienia takie jak wykrywanie i reagowanie na warunki przepięcia i zapadu napięcia, które mogą się zdarzać w skali mikrosekundowej. Konieczne jest także wykrywanie niebezpiecznych prądów i temperatur, rejestrowanie danych telemetrycznych i statusu urządzeń oraz dostarczanie tych informacji dalej do usługi debugowania, aby ułatwić życie użytkownikom i serwisantom danego sprzętu. Wszystkie te rozwiązania wymagają odpowiednio sprzętu analogowego oprócz samych cyfrowych układów programowalnych, sterujących obwodami zasilania.

Dla projektantów niebojących się wyzwań, którzy chcą podążać tą ścieżką, firmy takie jak Analog Devices zapewniają układy scalone,



**Rysunek 3. Schemat sekwencera zasilania współpracującego z pomocniczym układem FPGA oraz programowalnym kontrolerem napięcia LTC2936**

które pomagają w tym zadaniu. Są to na przykład programowalne komparatory. Układy takie jak LTC2936 zapewniają bardzo dokładne komparatory analogowe z programowalnym progami zadziałania, a także dodatkowymi progami do detekcji szybkich zdarzeń, które także są raportowane do układu sterującego. Układy tego typu są najczęściej wyposażone w pamięć nieulotną, dzięki czemu system działa natychmiast po załączeniu zasilania. Zwykle są wyposażone w interfejsy sterujące takie jak I<sup>2</sup>C lub SMBus, które pozwalają na odczyt i zapis danych konfiguracyjnych w pamięci układu, a także na dostęp do archiwalnych danych telemetrycznych, przechowywanych w układzie w celu debugowania. Na **rysunku 3** zaprezentowano przykładową aplikację układu LTC2936 do kontroli wielu domen zasilania. Oprócz funkcji szybkiego komparatora w układzie musi znaleźć się przetwornik analogowo-cyfrowy (ADC) do gromadzenia danych telemetrycznych. Sprawdzonym wyborem jest np. LTC2418, który pozwala na monitorowanie do 16 sygnałów analogowych. Układ ma precyzyjny, 24-bitowy przetwornik ADC. Do układu sterującego podłączany jest za pomocą interfejsu SPI.

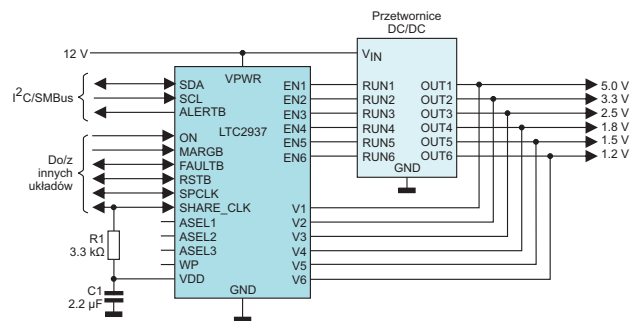
Podsumowując – istnieje wiele opcji stosowania FPGA lub CPLD do sterowania sekwencjonowaniem zasilania. To podejście działa, jednakże wymaga posiadania szerokich umiejętności, nie tylko projektowania systemów zasilania, ale także układów cyfrowych i tworzenia kodu dla układów programowalnych. Jest to poprawne i wartościowe podejście do projektowania tej sekcji układu, ale z pewnością nie jest ono najprostsze...

### Scalone sekwencery zasilania

Prostszym rozwiązaniem niż opisane powyżej jest zastosowanie scalonych kontrolerów zasilania. Z jednej strony oferują one precyzyjne i powtarzalne sekwencjonowanie zasilania oraz obsługę błędów w systemie zasilania. Z drugiej strony są znacznie prostsze w implementacji niż tworzenie samodzielnie systemu sterowanego z wykorzystaniem niewielkiego układu programowalnego.

Układy te realizują ważną funkcję sekwencjonowania szyn zasilających i upewnienia się, że podczas pracy napięcia zasilające pozostają w określonych granicach. Przykładem takiego elementu jest np. LTC2928. To łatwy w implementacji, konfigurowalny sekwencer zasilania z konfigurowalnymi progami napięcia podczas pracy. Może bez problemu uruchamiać sekwencje załączania i wyłączania układu, a także nadzorować napięcia podczas pracy, jednakże nie oferuje możliwości cyfrowego programowania ani telemetrii.

Na rynku dostępne są także inne układy do kontroli zasilania układu FPGA. Przykładem jest LTC2937, którego przykładowa aplikacja zaprezentowana jest na **rysunku 4**. Jest w pełni cyfrowo programowalny, realizuje algorytmy sekwencjonowania oparte na czasie i zdarzeniach oraz może sekwencjonować i nadzorować dowolną liczbę zewnętrznych zasilaczy. Ma możliwość obsługi błędów zasilacza i rejestrowania stanu podczas awarii w tzw. czarnej skrzynce w pamięci EEPROM. Układ tej klasy to doskonale rozwiązanie w przypadkach, gdzie cyfrowe



**Rysunek 4. Schemat układu do sekwencjonowania i nadzorowania napięcia zasilania, opartego na LTC2937**

zarządzaniem napięciem i telemetrią nie są ściśle wymagane, ale chcemy mieć większy poziom kontroli.

### Kompletny system zarządzania zasilaniem

Jeśli powyższe rozwiązania są nadal zbyt mało rozbudowane, jak na nasze potrzeby, możemy sięgnąć po scalone kontrolery zasilania PSM (*Power System Manager*). Układy tego rodzaju integrują w sobie szereg zaawansowanych funkcjonalności, oprócz opisanych powyżej możliwości sekwencjonowania i kontrolowania linii zasilania.

Aby w pełni wykorzystać zalety kompletnego PSM, można zastąpić jeden ze scalonych kontrolerów, jakie produkowane są przez większość wiodących producentów układów zasilania na rynku (np. Analog Devices, Maxim Integrated czy Texas Instruments). Wybór konkretnego modułu kontroli zasilania podyktowany jest najczęściej typem wybranego FPGA, liczbą szyn/domen zasilania itp.

Dzięki zastosowaniu scalonych PSM system uzyskuje możliwość autonomicznego sekwencjonowania zasilania podczas wyłączania i włączania, dokładnego kontrolowania napięcia poszczególnych szyn z dokładnością do 0,5% (lub nawet 0,25% w niektórych przypadkach) oraz mierzenia napięcia, prądu, temperatury i stanu poszczególnych modułów. Układy te potrafią pracować z bardziej złożonymi scenariuszami usterek i zapisywać szczegółowe informacje o awarii w pamięci EEPROM.

Seqwencjonowanie domen zasilania odbywa się za pomocą systemu uzgadniania czasu – wszystkie układy scalone zgadzają się na pewien czas zero i podstawę czasu, a następnie wszystkie zdarzenia w sekwencji mają miejsce w zaprogramowanym czasie (seqwencjonowanie oparte na czasie). Pozwala to na autonomiczną sekwencję napięć zasilania tak podczas załączania, jak i wyłączania systemu.

Grupa układów PSM obejmuje sterowniki z własnymi przełącznikami i analogową kontrolą pętli do obsługi przełączania zasilania we wszystkich aspektach. Alternatywnie, dostępne są menedżery zasilacza zawierające pętlę sprzężenia, która otacza zewnętrzny zasilacz, dodając do niego wszystkie funkcje zarządzania, w tym sekwencjonowanie, a także nadzór i monitorowanie, dla dowolnej szyny zasilania. Działa to niezależnie od architektury układów zasilających – moduły te współpracują zarówno z przetwornicami impulsowymi, jak i stabilizatorami LDO. Na **rysunku 5** pokazano aplikację przykładowego scalonego PSM typu LTC2974, sterującego czterema liniami zasilania.

### Układy µModule

Najbardziej ściśle zintegrowanymi rozwiązaniami, zapewniającymi największą funkcjonalność na centymetr kwadratowy, są urządzenia PSM typu µModule (tzw. mikromoduły) dostępne w obudowach BGA lub LGA. Są to kompletne systemy zasilania zintegrowane w jednym układzie scalonym. Znajdziemy w nim układy kontrolera, cewki indukcyjne, klucze/tranzystory przełączające i kondensatory. Niektóre stabilizatory typu µModule, takie jak na przykład LTM4650, nie zawierają funkcji cyfrowych, więc mogą skorzystać z sekwencjonowania i zarządzania z układów takich jak LTC2975. Inne µModule, takie jak LTM4676A, zawierają własne funkcje sterowania systemem zasilania

i można je łatwo zintegrować z innymi układami scalonymi pełniącymi funkcję PSM w systemie zasilania układu FPGA.

**Współdzielone sekwencjonowanie zasilania**

Mikromoduły PSM, układy menedżerów zasilania i scalone kontrolery stabilizatorów napięcia – wszystkie te elementy współpracują ze sobą podczas sekwencjonowania napięcia w czasie włączania i wyłączenia systemu. Aby było to możliwe, używają jednoprzewodowej magistrali do współdzielenia informacji i synchronizacji w czasie sekwencjonowania. Magistrala ta nazywana jest SHARE\_CLK. Za pomocą tego pojedynczego przewodu wszystkie układy scalone w zasilaczu współdzielą informacje o tym, kiedy należy rozpocząć sekwencjonowanie (czas zero), kiedy następuje każde tyknięcie zegara, a także przesyłają inne informacje o stanie zasilacza, które wpływają na sekwencjonowanie (na przykład podczas awarii).

W systemie wystarczy po prostu połączyć wszystkie piny SHARE\_CLK poszczególnych układów, aby umożliwić koordynację ich działań. Każdy układ scalony ma własne, programowalne elementy taktowania sekwencyjnego, ale może korzystać ze wspólnej podstawy czasu w celu dokładnego i niezawodnego pomiaru zdarzeń, takich jak włączanie i wyłączenie stabilizatorów, narastanie napięcia i przekroczenie limitu czasu w przypadku awarii.

Pin SHARE\_CLK, na najbardziej podstawowym poziomie działania, jest pinem zegarowym typu otwarty dren, oferującym sygnał 100 kHz. Charakter otwartego drenu oznacza, że układ scalony może albo aktywnie ściągnąć sygnał do masy, albo pozwolić na stan wysoki. Kiedy wszystkie urządzenia w magistrali odpuszczają ściągnięcie magistrali, rezystory podciągające podnoszą napięcie do poziomu 3,3 V. Pozwala to jednemu urządzeniu zatrzymać zegar, ściągnąć go w dół, aż będzie gotowe. Oznacza to, że wszystkie urządzenia muszą się zgodzić, zanim zegar zacznie realnie działać; jest to skuteczny mechanizm przekazywania zegara sekwencjonowanego, a także wskazywania statusu np. poprzez zatrzymanie magistrali.

**Współdzielona obsługa błędów i awarii**

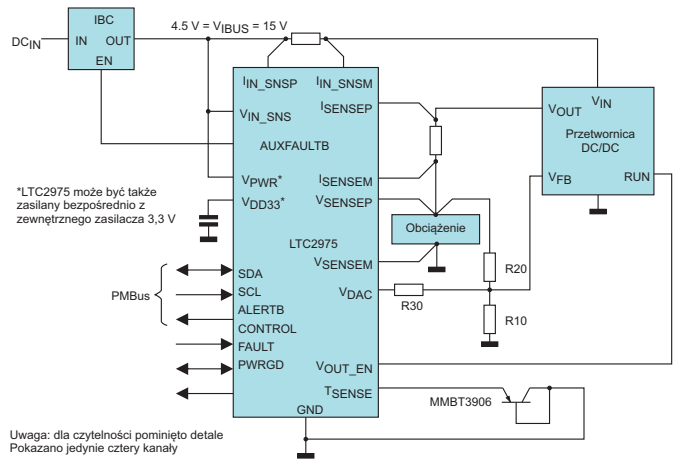
Podobna do pinu SHARE\_CLK jest szyna FAULT. Każdy układ w systemie jest podłączony do wspólnego przewodu FAULT i może ściągnąć go do masy – urządzenia w magistrali FAULT mają wejścia typu otwarty dren. Dzięki temu, podobnie jak w przypadku SHARE\_CLK, pojedynczy układ może ustawić linię na potencjalne masy, tak by inne mogły zareagować na wystąpienie awarii. Zapewnia to prosty i szybki sposób komunikacji całej gamie urządzeń PSM i możliwość reagowania na awarie innych układów. Zachowanie to jest w pełni konfigurowalne i pozwala na skoordynowaną reakcję CAŁEGO SYSTEMU, gdy coś pójdzie nie tak, podczas sekwencjonowania lub w czasie pracy zasilacza.

System można skonfigurować tak, aby podczas awarii odłączał zasilanie od systemu i podejmował próbę ponownego sekwencjonowania zgodnie z określonym harmonogramem, rejestrując jednocześnie dane do tzw. czarnej skrzynki. Jest to pamięć EEPROM, podłączona do układów poprzez interfejs I<sup>2</sup>C, na które zapisywane są dane o stanie systemu i przyczynach usterki, która nastąpiła. System lub serwisant może następnie uzyskać dostęp do tych danych poprzez I2C w celu analizy przyczyn awarii i łatwiejszego usunięcia przyczyn usterki z systemu.

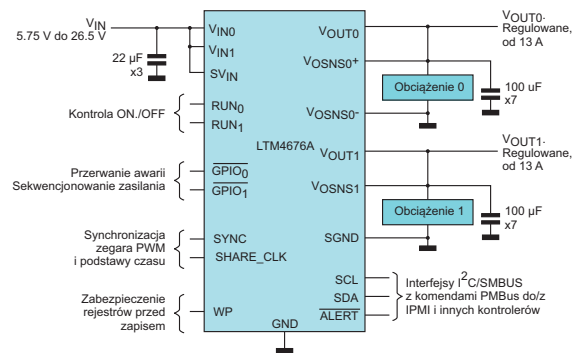
**Sekwencja wyłączenia zasilania – co zrobić ze zgromadzoną energią?**

Podczas sekwencjonowania domen zasilania należy wziąć pod uwagę dodatkowo kwestie związane z zarządzaniem energią w systemie. Jest to konieczne, aby zapewnić deterministyczne czasy załączania i wyłączenia zasilaczy. Zarządzanie energią zgromadzoną w systemie, zwłaszcza w czasie wyłączania linii zasilania, jest krytyczne dla uzyskania deterministycznego czasu odłączania poszczególnych domen zasilania.

W typowym zasilaczu znaleźć można dziesiątki dużych kondensatorów elektrolitycznych magazynujących ładunek na wyjściu zasilania. Są naładowane do napięcia zasilania i utrzymują wystarczającą ilość energii, aby uszkodzić podłączone do nich systemy, jeżeli nie



**Rysunek 5. Schemat czterokanałowego systemu zarządzania zasilaniem opartego na kontrolerze LTC2975**

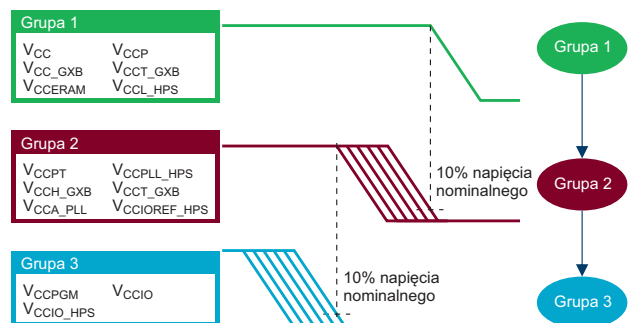


**Rysunek 6. Schemat aplikacji µModule LTM4676A z dwoma domenami zasilania o prądzie wyjściowym 13 A**

zostaną odpowiednio rozładowane w czasie sekwencji wyłączania zasilania. Aby tego uniknąć, producenci układów programowalnych określają odpowiednie sekwencje wyłączania, która chroni urządzenie. Dla Altera Arria 10 sekwencja wygląda jak na **rysunku 7**. W opisanej sekwencji, trochę pomiędzy wierszami, znajduje się informacja, że cała energia zgromadzona w kondensatorach filtrujących poszczególne linie zasilania musi zostać bezpiecznie rozproszona.

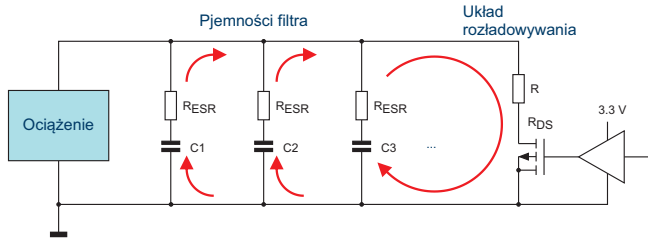
Najprostszy sposób na rozproszenie mocy pojemności filtrujących to stały rezystor pomiędzy linią zasilania a masą. Rezystor ten zawsze rozprasza moc, gdy zasilanie jest włączone. Jego rezystancja musi być wystarczająco duża, by straty podczas normalnej pracy systemu były minimalne, ale także na tyle mała, by stała czasowa τ rozładowania była akceptowalnie krótka. Czas potrzebny do wystarczającego rozładowania zasilania przyjmuje się jako wielokrotność (zwykle 5x) stałej czasowej RC. Czas ten powinien być zoptymalizowany, tak aby moc statyczna rozproszona w rezystorze była akceptowalna (na przykład

**Wymagania sekwencji wyłączania**

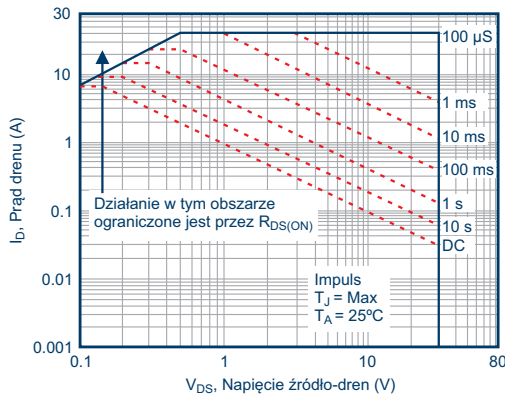


**Rysunek 7. Sekwencja odłączania napięć zasilania dla Altera Arria 10**





**Rysunek 8. Rozładowywanie pojemności zasilacza za pomocą tranzystora FET**



**Rysunek 9. Bezpieczny obszar pracy tranzystora MOSFET z kanałem typu N**

<0,25 W). Dla pojemności na poziomie 1 mF i napięcia zasilania 1,0 V, opornik o rezystancji równej 4 Ω będzie miał stałą czasową rozładowywania układu  $\tau=4$  ms i rozładuje zasilanie do napięcia poniżej 50 mV w czasie zaledwie około 13 ms. Takie podejście jest wystarczające, o ile zastosowany rezystor ma moc równą co najmniej 0,25 W, a system działa ze stałą stratą 0,25 W i czasem rozładowania 13 ms.

Bardziej złożoną, ale również bezpieczniejszą opcją jest dołączenie rezystora na kondensatorze tylko wtedy, gdy nadszedł czas na rozładowanie zasilania. Takie podejście pomaga szybciej usunąć ładunek z kondensatorów, gdy nadejdzie odpowiedni czas. Pozwala to na bezpieczne rozładowanie pojemności na oporniku i tranzystorze polowym (FET), a jednocześnie eliminuje ciągłe straty mocy, jak w przypadku rezystora podłączonego do układu na stałe. Przykład takiego obwodu zaprezentowano na **rysunku 8**. Podejście wykorzystujące tranzystor do sterowania układem rozładowywania pojemności wymaga rozważenia kilku czynników w układzie. Konieczne jest przeanalizowanie sposobu kontroli klucza, czasu rozładowania i oszacowanie rozpraszanej mocy.

Do sterowania tego rodzaju tranzystorem konieczny jest sygnał o odpowiednio wysokim napięciu, nakazujący zamknięcie (zwarcie) klucza w odpowiednim momencie. Przełącznik FET to tranzystor MOS z kanałem typu N, więc sygnał sterujący musi wzrosnąć powyżej  $V_{TH}$  tego elementu – wystarczająco, aby doprowadzić go do poziomu nasycenia. W przypadku typowych tranzystorów polowych napięcie progowe bramki może wynosić od 3 V do 5 V.

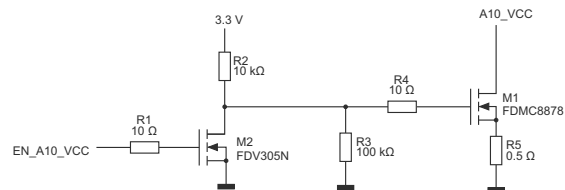
Typowy kondensator elektrolityczny filtrujący zasilanie będzie miał setki miliomów równoważnej rezystancji szeregowej (ESR), co rozproszy część energii podczas rozładowywania tego elementu, ale zazwyczaj w układzie znajduje się wiele takich kondensatorów połączonych równolegle. Oznacza to, że całkowita pojemność baterii kondensatorów może sumować się do kilkudziesięciu milifaradów, a równoważny opór szeregowy wyniesie dziesiątki miliomów lub nawet mniej. Bezpiecznie jest założyć, że ESR kondensatorów rozproszy jedynie niewielką część zmagazynowanej w nich energii. Aby rozładować pojemności w rozsądnym czasie, stała czasowa RC układu rozładowania musi być mniejsza niż 1/5 pożądanego czasu rozładowania baterii kondensatorów filtrujących (tak aby napięcie spadło odpowiednio szybko poniżej poziomu kilku miliwoltów). Jest to proste do obliczenia (**równanie 1**) z użyciem sumy pojemności wszystkich

kondensatorów oraz sumy rezystancji tranzystora kanału ( $R_{DS}$ ) i szeregowej rezystancji R, a także równoległej sumy rezystancji  $R_{ESR}$ . Parametr N jest liczbą połączonych równolegle ze sobą kondensatorów w układzie filtrującym napięcie.

$$\tau = \left( \frac{R_{ESR}}{N} + R + R_{FET} \right) \cdot (C_1 + C_2 + \dots + C_N) \quad (1)$$

W przypadku typowego dużego systemu zasilania z baterią kondensatorów o łącznej pojemności 50 mF i sumą  $R_{DS}+R=500$  mΩ napięcie spadnie poniżej 50 mV w około 125 ms. Prąd szczytowy rozładowania w tym czasie wynosi  $1/500$  mΩ=2 A, co przy napięciu 1 V daje moc maksymalną 2 W. Tyle najwięcej ciepła wydzieli się na oporniku R i tranzystorze polowym. Należy pamiętać, że większość zmagazynowanej energii jest wydzielenia w czasie pierwszych dwóch stałych czasowych układu, zatem dobierając tranzystor, można skorzystać z jego charakterystyk impulsowych, a nie DC. Na tej podstawie można zdecydować, czy rezystor szeregowy R jest w systemie konieczny. Wynika to z wykresu bezpiecznych obszarów pracy wybranego tranzystora FET, takiego jak na **rysunku 9**. Jak wynika z tej zależności wybrany FET bezpiecznie wytrzyma impuls o mocy 2 W i czasie powyżej 10 sekund, więc nie ma niebezpieczeństwa jego uszkodzenia. Ten tranzystor FET charakteryzuje się jednak rezystancją  $R_{DS}$  mniejszą niż 20 mΩ, dlatego szeregowy opornik R jest wymagany i musi mieć rezystancję równą co najmniej 480 mΩ. Rezystor musi być dobrany tak, aby poradził sobie z wydzielanym ciepłem, ponieważ to on rozproszy większość mocy. Pamiętać trzeba, że czas trwania impulsu będzie znacznie krótszy niż termiczna stała czasowa opornika, którą znaleźć można w karcie katalogowej tego elementu.

Niezawodny obwód rozładowania domen zasilania powinien bezpiecznie rozpraszać energię w różnych warunkach. Obwód z **rysunku 10** to sprawdzona metoda na rozładowanie energii zgromadzonej w pojemnościach systemu. Zastosowano tranzystor MOSFET FDMC8878 firmy ON i duży rezystor (w obudowie 1210 do montażu powierzchniowego) o rezystancji 0,5 Ω.



**Rysunek 10. Układ rozładujący z tranzystorem MOSFET**

## Podsumowanie

W powyższej, drugiej części cyklu dotyczącego zasilania układów programowalnych opisaliśmy, wraz ze wszystkimi niuansami, dynamiczne zachowanie napięcia podczas włączania i wyłączenia. Teraz każdy czytelnik powinien nie tylko rozumieć potrzebę sekwencjonowania napięć zasilania układu FPGA, ale także wiedzieć, jak to zrobić i na co zwrócić uwagę, projektując tego typu system. Przykłady realnych implementacji układów do sekwencjonowania domen zasilania doskonale uzupełniają teoretyczny opis mechanizmów sekwencjonowania.

W pierwszych dwóch częściach odpowiedzieliśmy na pytanie „jak?” dotyczące systemów zasilania nowoczesnych układów FPGA. W ostatniej, trzeciej części, uzupełnimy powyższe informacje o odpowiedzi na pytanie „czemu?”, sięgając do fizycznych podstaw budowy współczesnych układów programowalnych.

**Nikodem Czechowski, EP**

## Źródła:

1. <http://bit.ly/35D2nPm>
2. „Intel Arria 10 Core Fabric and General Purpose I/Os Handbook”. Intel, August 2018.
3. „Virtex UltraScale FPGAs Data Sheet: DC and AC Switching Characteristics”. Xilinx, January 2018.