

# EtherCAT w praktyce

## Zastosowanie mikrokontrolera Sitara do implementacji EtherCAT

Protokół komunikacyjny EtherCAT jest wśród wiodących standardów komunikacyjnych, używających do transmisji danych sieci Ethernet. Jest on coraz powszechniej stosowany w środowisku przemysłowym dla urządzeń wejścia/wyjścia, czujników i sterowników programowalnych PLC. EtherCAT został opracowany przez niemiecką firmę Beckhoff Automation, a później jego standaryzacją zajęła się grupa EtherCAT Technology Group (ETG), utworzona w celu pomocy w rozpowszechnianiu standardu. Aktualnie ponad 1700 firm z 52 krajów wytwarza produkty komunikujące się za pomocą tego protokołu.

Interfejs Ethernet został zaadoptowany równolegle w różnorodnych aplikacjach, jednak w środowisku przemysłowym nie jest zbyt efektywnym medium do wymiany niewielkich ilości danych, raczej umiarkowanie nadaje się dla operacji wykonywanych w czasie rzeczywistym i pracuje tylko w topologii gwiazdy, w której węzły sieci muszą być dołączone za pomocą rozdzielaczy. Technologia EtherCAT dodaje kilka znaczących ulepszeń do oryginalnego Ethernetu i dodatkowe topologie sieci, dając możliwość utworzenia efektywnej sieci do zastosowania w automatyce przemysłowej, w pełni zgodnej ze specyfikacją Ethernetu.

Specyfika EtherCAT pozwala standardowemu komputerowi PC być używanym jako zarządca – EtherCAT master i komunikować się z urządzeniami podrzędnymi, zgodnymi z wymaganiami protokołu EtherCAT – EtherCAT slave. Protokół EtherCAT może być używany we wszystkich urządzeniach pracujących w sieci przemysłowej – kontrolerach procesów, interfejsach operatorów, zdalnych urządzeniach wejść/wyjść, czujnikach, aktuatorach, napędach i innych.

EtherCAT wspiera wszystkie topologie sieciowe: magistralę, gwiazdową lub drzewiastą – wspólne struktury magistral sieci przemysłowych również mogą być wykonywane za pomocą EtherCAT. Ponieważ port interfejsu EtherCAT jest dostępny w urządzeniach I/O, nie ma wymagania stosowania urządzeń przełączających. Korzystając z łączy kablowych o długości do 100 m oraz dłuższych, optycznych, sieć EtherCAT może rozciągać się na przestrzeni tysięcy urządzeń rozproszonych na dużym obszarze geograficznym. Na krót-

kie odległości, jak na przykład połączenia w obrębie płyty montażowej, EtherCAT używa E-bus, technologii różnicowego przesyłania sygnałów. Przykładową strukturę sieci EtherCAT pokazano na rysunku 1.

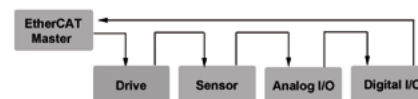
Dzięki implementacji funkcji przetwarzania danych „w locie”, EtherCAT poprawia własności typowej sieci Ethernet. Węzły sieci EtherCAT odczytują dane z ramek podczas ich odbierania. Wszystkie ramki EtherCAT pochodzą od zarządcy sieci (EtherCAT master), który przesyła komendy i dane do urządzeń slave. Jakikolwiek dane przesyłane z powrotem do mastera są zapisywane przez slave do przechodzących ramek danych. Pomaga to w wyeliminowaniu potrzeby wymiany niewielkich ramek danych w konfiguracji punkt-punkt, pomiędzy urządzeniem master i poszczególnymi urządzeniami slave, co ogromnie poprawia efektywność komunikacji w sieci. Z drugiej strony, oznacza to, że każdy slave musi mieć dwa porty Ethernet, musi być „przezroczystym” dla obieranej transmisji i być w stanie odczytywać i zapisy-

wać dane do aktualnie przesyłanej ramki. Z tego powodu jest wymagane zastosowanie specjalizowanych układów interfejsowych. W rezultacie modyfikacji, użyteczne pasmo w sieci o przepustowości 100 Mb/s pracującej z wykorzystaniem EtherCAT wynosi więcej niż 90% prędkości transmisji i jest porównywalne do mniej niż 5% ogółu znanych standardów sieci, w których master musi indywidualnie komunikować się z każdym węzłem slave.

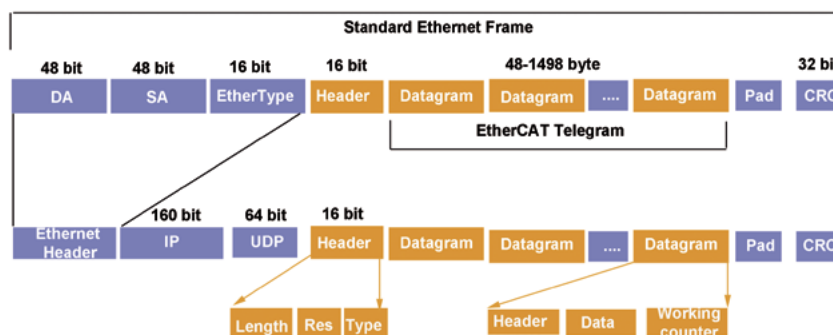
### Telegram EtherCAT

Jako pokazano na rysunku 2, telegram EtherCAT jest inkapsulowany w ramce Ethernet i zawiera jeden lub więcej datagramów EtherCAT dostarczanych do urządzeń slave. Ramki protokołu EtherCAT są oznaczane w nagłówku ramki Ethernet lub mogą one być upakowane w nagłówku ramki IP/UDP. Gdy jest używany nagłówek IP, to transmisja EtherCAT może być przesyłana poprzez typowe routery sieciowe.

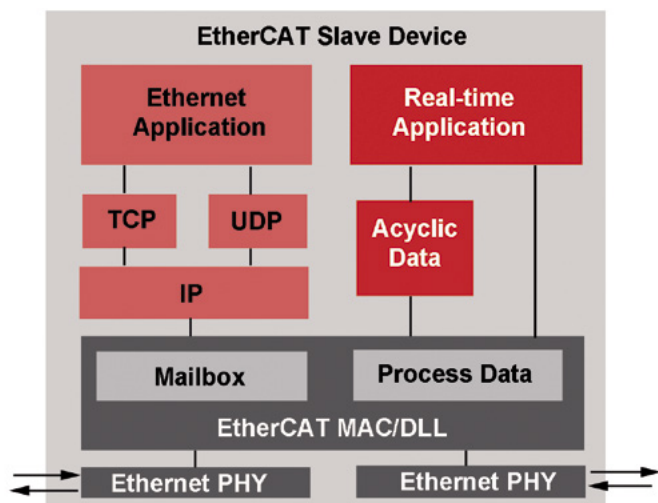
Każdy datagram EtherCAT jest komendą, która zawiera nagłówek, dane i licznik roboczy. Nagłówek i dane są używane do wyspecyfikowania operacji, które muszą być wykonane przez slave, natomiast licznik roboczy jest aktualizowany przez slave informując mastera, że slave odebrał i przetwarza komendę.



Rysunek 1. Przykład sieci EtherCAT



Rysunek 2. Telegram EtherCAT



Rysunek 3. Komponenty składowe węzła EtherCAT

### Protokół

Każdy *slave* przetwarza pakiety otrzymane od *master'a* „w locie” odbierając ramkę, dokonując oceny jej zawartości i podejmując odpowiednią akcję, jeśli adres podany w datagramie EtherCAT zgadza się z jego własnym adresem. W międzyczasie wysyła cały datagram za pomocą drugiego portu modyfikując zawartość pola sumy kontrolnej CRC pakietu. Za pomocą datagramu *master* adresuje całkowitą przestrzeń adresową o wielkości do 4 GB, w której może być do 65536 urządzeń *slave*, z których każde może mieć obsadzone do 65536 adresów. Datagram EtherCAT nie ma żadnych ograniczeń w kolejności, w której urządzenia *slave* są adresowane w odniesieniu do rzeczywistego położenia (kolejności) węzła w sieci.

W sieciach EtherCAT występują różne typy transmisji danych – cykliczne i niecykliczne. Dane cykliczne są danymi procesowymi, które są przesyłane okresowo w pewnych interwałach, jednym słowem – cyklicznie. Zazwyczaj dane niecykliczne są danymi niekrytycznymi, które mogą mieć sporą objętość i są przesyłane w odpowiedzi na komendę kontrolera. Niektóre dane acykliczne, takie jak dane diagnostyczne, mogą być krytyczne znaczenie i przez to mieć ściśle określone wymagania czasowe. EtherCAT obsługuje te inne wymagania odnośnie do transmisji danych poprzez zoptymalizowany schemat adresowania – adresowanie fizyczne, logiczne, wielokrotne oraz jednoczesne rozsyłanie komunikatów do wszystkich urządzeń (*broadcast*).

Aby obsłużyć odmienne schematy adresowania, każdy *slave* ma pamięć wspomagającą zarządzanie komunikacją sieciową (*FMMU*, *Fieldbus Memory Management Unit*). Pamięć ta w każdym urządzeniu *slave* pozwala na używanie protokołu EtherCAT w taki sposób, aby traktować różne urządzenia *slave* jako części wielkiego, pojedynczego obszaru adresowania o wielkości 4 GB z przestrzeniami adresowymi poszczególnych urządzeń mapowanymi w jego obrębie. *Master* przygotowuje kompletny obraz procesu w fazie inicjalizacji, a następnie umożliwia dostęp do każdego bitu w obszarze adresowania *slave* za pomocą pojedynczej komendy EtherCAT. Ta właściwość pozwala na komunikowanie się z każdą liczbą kanałów wejść/wyjść w złożonych i nieskomplikowanych urządzeniach połączonych ze sobą za pomocą standardowego kontrolera Ethernetu i standardowych kabli Ethernet.

### Efektywność sieci

Jako rezultat zastosowania sprzętowej jednostki FMMU oraz przetwarzania „w locie”, sieć EtherCAT pracuje z bardzo dużą efektywnością. Pozwala to na uzyskanie czasu komunikacji rzędu mikrosekund mierzonego od kontrolera do urządzenia zainstalowanego na obiekcie. Efektywność komunikowania się nie jest wąskim gardłem w sieciach EtherCAT i współgra z prędkością obliczeniową

współczesnego komputera przemysłowego PC. W ten sposób staje się możliwe np. kontrolowanie poprzez sieć napędu zadajnika z czujnikiem położenia mającym interfejs w postaci pętli prądowej.

### Topologia

Standard EtherCAT wspiera wszystkie topologie sieciowe – magistralę, gwiazdę lub drzewo. Struktury magistral transmisyjnych innych sieci przemysłowych, również mogą być wykonane za pomocą EtherCAT. Ponieważ wyjście interfejsu EtherCAT jest dostępne w urządzeniach dołączonych do sieci, to nie ma wymagania stosowania urządzeń przełączających. Korzystając z łączy kablowych o długości do 100 m oraz dłuższych, optycznych, EtherCAT może rozciągać się na przestrzeni tysięcy urządzeń rozproszonych na dużym obszarze geograficznym. Na krótkie odległości, jak na przykład połączenia w obrębie płyty montażowej, EtherCAT używa E-bus, technologii różnicowego przesyłania sygnałów.

### Dystrybuowany zegar

Aby umożliwić równoczesną realizację zadań w węzłach przemysłowych zainstalowanych w pewnej odległości od siebie, niezbędna jest synchronizacja ich zegarów wewnętrznych. EtherCAT umożliwia to dzięki próbkowaniu znaczników gdy pakiet dociera i opuszcza każdy węzeł *slave* podczas swojej wędrówki przez sieć. *Master* używa informacji zawartej w znacznikach czasu dostarczonych przez poszczególne urządzenia *slave*, aby dokładnie wyliczyć opóźnienie propagacji dla poszczególnych urządzeń współpracujących w sieci. W każdym węźle *slave* zegar jest regulowany na bazie tych obliczeń i w ten sposób wszystkie zegary są synchronizowane co 1  $\mu$ s. Dodatkową zaletą dokładnej synchronizacji zegarów jest to, że wszystkie pobierane pomiary mogą być odnoszone do zsynchronizowanego czasu i jest dzięki temu eliminowana niepewność skojarzona z jitterem komunikacyjnym pomiędzy urządzeniami.

### Profile urządzeń

W automatyce przemysłowej używanie profili jest ogólnie stosowaną metodą służącą do opisu dostępnej funkcjonalności i parametrów urządzeń. Specyfikacja EtherCAT nie wprowadza żadnych nowych profili urządzeń, ale zamiast tego są dostarczane interfejsy dla już istniejących profili, dzięki czemu urządzenia mogą być łatwo zaktualizowane i włączone do sieci EtherCAT. Niektóre z tych interfejsów to np. *CANOpen over EtherCAT* i *SERCOS over EtherCAT*, które pozwalają na używanie zalet mapowania ich struktur danych przez EtherCAT.

### Komponenty węzła EtherCAT

Każdy węzeł EtherCAT ma trzy komponenty: warstwę fizyczną, warstwę danych i warstwę aplikacji. Warstwę fizyczną zaimplementowano z użyciem kabla 100BASE-TX, światłowodu 100BASE-FX lub magistrali E-bus pracującej z wykorzystaniem interfejsu różnicowego LVDS. Interfejs MAC zaimplementowano z użyciem specjalizowanego układu ASIC lub FPGA zgodnie ze specyfikacją standardu EtherCAT. Poza MAC jest aplikacja przemysłowa, która „troszczy się” o funkcjonowanie urządzenia i standardowy stos TCP/IP oraz UDP/IP, aby wspierać urządzenia bazujące na profilu Ethernet. Zależnie od stopnia złożoności urządzenia, węzeł EtherCAT może być zaimplementowany sprzętowo lub może być kombinacją sprzętu i oprogramowania pracującego we wbudowanym procesorze.

### Typowy węzeł EtherCAT

Typowy węzeł EtherCAT, który jest używany współcześnie, ma architekturę zbliżoną do tej pokazanej na **rysunku 3**. Wiele nieskomplikowanych urządzeń EtherCAT, takich jak np. moduły cyfrowych wejść/wyjść, może być zbudowanych z użyciem dostęp-



Rysunek 4. Urządzenie wejść/wyjść cyfrowych z interfejsem EtherCAT

nych układów FPGA lub ASIC. Uproszczoną wersję takiej architektury przedstawiono na **rysunku 4**. Jest ona dobrze dopasowana do użycia w niedrogich, nieskomplikowanych węzłach wejść/wyjść, nie wymaga dodatkowego oprogramowania nadzorującego pracę, a cała funkcjonalność jest zaimplementowana sprzętowo.

W węzłach EtherCAT, w których jest wymagana dodatkowa moc obliczeniowa, zewnętrzny procesor (zwykle wyposażony w pamięć Flash) jest dołączony do układów EtherCAT ASIC/FPGA w celu umożliwienia obsługi na poziomie aplikacji. Takie rozwiązanie może być wymagane np. dla czujnika, w którym jest konieczne użycie mikroprocesora od odczytu danych z sensora, implementacji sterownika i uruchomienia stosu aplikacji EtherCAT. Koszt takiej architektury jest wyższy, niż dla nieskomplikowanych urządzeń I/O. Jest on oferowany z elastycznością pozwalającą konstruktorowi na podjęcie decyzji odnośnie do wyboru mikroprocesora, który najlepiej spełnia wymagania odnośnie do realizowanych zadań oraz ceny.

W innym ujęciu, implementacja EtherCAT jest jednym z układów peryferyjnych w urządzeniu, które ma zintegrowane CPU. Wiele układów FPGA ma możliwość zdefiniowania rdzenia procesora w FPGA lub już ma zintegrowany procesor. Niektórzy dostawcy dostarczają układy ASIC z EtherCAT i odpowiednim procesorem w układzie. Układy FPGA są bardzo elastyczne w aplikacjach, ale zależnie od wybranego CPU istnieje ryzyko, że koszt implementacji lub częstotliwość taktowania będą, odpowiednio, zbyt duże lub zbyt małe i staną się wyzwaniem dla przewyższenia.

### EtherCAT w pojedynczym układzie scalonym

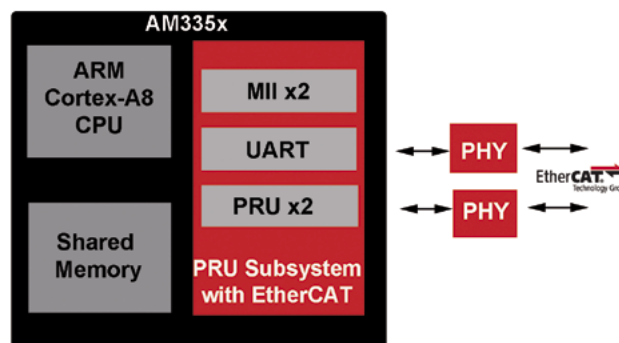
Podczas gdy opisane wcześniej architektury pracują perfekcyjnie z zaimplementowanym protokołem EtherCAT, wymagają one dwóch układów scalonych, co zwiększa wymagania odnośnie do powierzchni oraz podnosi koszt komponentów dla opracowania przemysłowego. Rozwiązanie jednoukładowe integruje rdzeń Cortex-A8 z wieloma układami peryferyjnymi i interfejsami. Jest ono dobrze dopasowane dla potrzeb opracowywania wyposażenia przemysłowego automatyki i może obniżyć koszty podzespołów o co najmniej 30%.



Rysunek 5. EtherCAT z układem ASIC i procesorem zewnętrznym



Rysunek 6. EtherCAT zintegrowany z procesorem



Rysunek 7. EtherCAT Slave w mikrokontrolerze Sitara AM335x z rdzeniem ARM

Zintegrowanie podsystemu programowalnej jednostki czasu rzeczywistego PRU (*Programmable Real-time Unit*), która ma bardzo niski poziom interakcji z interfejsem MII (*Media Independent Interface*), pozwala podsystemowi PRU na obsługę specjalizowanego protokołu komunikacyjnego, takiego jak EtherCAT. Cała warstwa EtherCAT jest inkapsulowana w podsystemie PRU, w jego firmwarze. PRU przetwarza telegramy EtherCAT „w locie”, interpretuje je, dekoduje adres i wykonuje komendy EtherCAT. Dla celu komunikowania się stosu EtherCAT (warstwa 7) oraz aplikacji przemysłowej z rdzeniem ARM są używane przerwania.

Podsystem PRU wykonuje również przekazywanie ramek w odwrotnym kierunku. Ponieważ podsystem PRU obsługuje funkcjonalność EtherCAT, procesor ARM może być używany do realizacji złożonych aplikacji, a w nieskomplikowanych aplikacjach można zastosować rdzeń o mniejszej częstotliwości taktowania w celu obniżenia ceny gotowego wyrobu.

### Architektura programu EtherCAT

Również oprogramowanie jest bardzo ważnym czynnikiem upewniającym, że implementacja EtherCAT w danym urządzeniu pracuje gładko i bez zacięć.

Konstruktor przy wyborze rozwiązania jednoukładowego musi rozważyć trzy główne kryteria. Pierwszym jest mikrokod, który implementuje funkcjonalność Warstwy 2 w PRU. Drugi to stos aplikacji *slave*, który jest uruchamiany na mikrokontrolerze ARM. Trzecim jest aplikacja przemysłowa, która jest zależna od wyposażenia używanego w danym opracowaniu.

Na **rysunku 7** pokazano wspomniane wcześniej rozwiązanie z pojedynczym układem scalonym Wykonano je w oparciu o mikrokontroler **Sitara AM344x ARM Cortex-A8** produkowany przez firmę Texas Instruments. Opracowanie jest przykładem rozwiązania i jednocześnie ilustruje, jak wiele urządzeń peryferyjnych może być zintegrowanych w pojedynczym układzie scalonym w celu implementacji obsługi protokołu EtherCAT. Dodatkowo do wymienionego wcześniej oprogramowania, są używane dodatkowe komponenty wspomagające, takie jak warstwa adaptacyjna protokołu i sterowniki urządzeń, dostarczane z zestawem deweloperskim firmy TI.

Możliwości opracowywania urządzeń automatyki przemysłowej z EtherCAT przy zastosowaniu rozwiązania jednoukładowego są nieskończone.

**Maneesh Soni**  
Systems Engineering Lead  
Sitara ARM microprocessors  
Texas Instruments