

Redakcja Elektroniki Praktycznej dziękuje firmie NDN z Warszawy, autoryzowanemu dystrybutorowi i serwisowi firmy Rigol, za udostępnienie odcyloskopu Rigol DS2202 dla potrzeb tego artykułu.

Analiza protokołów (3)

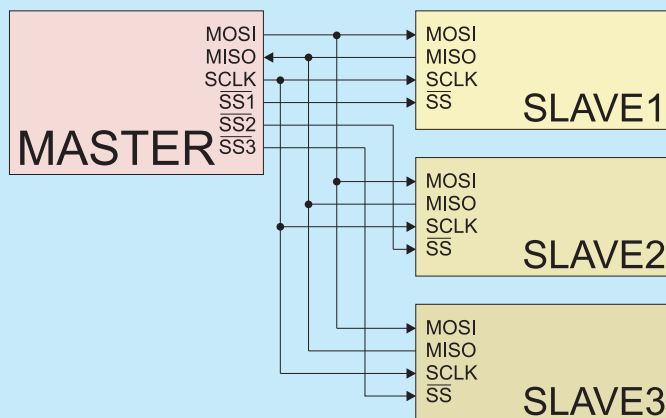
Analizowanie interfejsu SPI

Termin „analiza protokołów” znany jest elektronikom nie od dziś. Specjaliści różnych dziedzin interpretują go jednak zgoła odmiennie. Inżynier telekomunikacji zajmujący się systemami łączności radiowej czy telefonii komórkowej analizę protokołów będzie rozumiał inaczej niż informatyk, a jeszcze inaczej konstruktor projektujący układy elektroniczne. W artykule zajmiemy się protokołami wykorzystywanymi w popularnych interfejsach komunikacyjnych.

W poprzednich odcinkach poznaliśmy ogólną zasadę obsługi analizatora protokołów oraz metody badania protokołu RS-232/UART. Mieliliśmy do czynienia z transmisją asynchroniczną, różniącą się zasadniczo od transmisji synchronicznej stosowanej w interfejsie SPI. W konsekwencji konieczne stało się zwiększenie linii interfejsu. Dodano więc linię SCLK, wykorzystywaną do przesyłania przebiegu zegarowego taktującego wszystkie nadajniki i odbiorniki, a także linię (SS) służącą do sprzętowego wybierania urządzenia podrzędnego – Slave, komunikującego się z urządzeniem nadrzędnym – Master. Dane pojawiają się wprawdzie jednocześnie na wejściach wszystkich odbiorników, ale analizuje je tylko ten Slave, który został wybrany sygnałem SS. Biorąc pod uwagę zależności między poszczególnymi sygnałami można wyróżnić cztery tryby pracy interfejsu, którym musimy się bliżej przyjrzeć.

Tryby pracy interfejsu SPI

Urządzeń Slave może być w systemie kilka, ale w danej chwili Master prowadzi komunikację tylko z jednym, wybierając go podaniem niskiego poziomu na linię SS. Jeżeli Master ma komunikować się z kilkoma urządzeniami, musi mieć odpowiednią liczbę wyjść sterujących SS_i. Program dla Mastera musi być napisany tak, aby podczas transmisji, w danej chwili wyst-



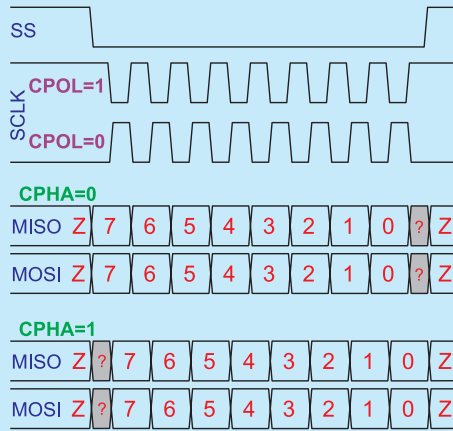
Rysunek 21. Przykładowe połączenia urządzeń komunikujących się ze sobą przez interfejs SPI

Tryb	CPOL	CPHA
0	0	0
1	0	1
2	1	0
3	1	1

wiać poziom aktywny tylko na jednym wyjściu SS_i. Po przesłaniu danych wszystkie linie sterujące powinny być ustawione w poziom nieaktywny (wysoki). Przykładowe połączenia urządzeń komunikujących się ze sobą przez interfejs SPI przedstawiono na **rysunku 21**. Jak widać z schematu, koncepcja interfejsu SPI nie przewiduje bezpośredniego komunikowania się ze sobą urządzeń Slave. Jeśli wymaga tego aplikacja, musi być stworzony odpowiedni protokół realizujący sekwencję połączeń np. Slave1->Master, Master->Slave2.

Wróćmy do budowy interfejsu SPI. Zawiera on co najmniej 4 linie. Dwie z nich już znamy. Są to: SCLK (Serial Clock) – przebieg zegarowy taktujący transmisją i SS (Slave Select) – sygnał wyboru urządzenia Slave (poziomem niskim). Dane są przesyłane liniami: MOSI (Master Output Slave Input) w kierunku od urządzenia Master do Slave oraz MISO (Master Input Slave Output) od urządzenia Slave do Master. Należy zwrócić uwagę na to, że w wyniku przyjęcia takiej konwencji wszystkie wyprowadzenia MOSI są łączone ze sobą, podobnie jest z wyprowadzeniami MISO oraz SCLK. W interfejsie RS-232 wyjście nadajnika TxD należało łączyć krzyżowo z wejściem odbiornika RxD.

Zanim zaczniemy pomiary należy jeszcze rozważyć pewien problem techniczny związany z taktowaniem danych na liniach MOSI i MISO. Interfejs SPI może pracować w kilku trybach definiowanych na podstawie polaryzacji i fazy przebiegu zegarowego. Możliwe warianty są definiowane umownymi parametrami CPOL i CPHA. Parametr CPOL decyduje o poziomie sygnału zegarowego SCLK, jaki jest utrzymywany podczas nieaktywności interfejsu, CPHA określa natomiast zbocza sygnału zegarowego, na których następuje zmiana poziomu na liniach MOSIO i MISO. Na przeciwnych zboczach odbiorniki odczytują dane



Rysunek 22. Możliwe tryby pracy interfejsu SPI

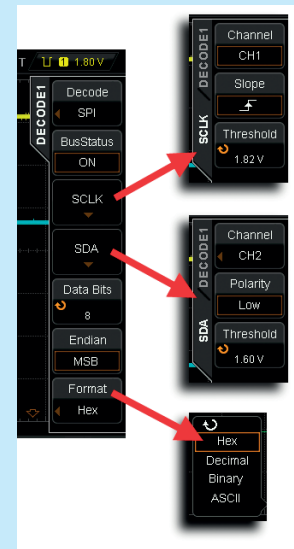
z tych linii. Możliwe kombinacje przedstawiono na **rysunku 22**. Odpowiadają im jednoznacznie określone tryby pracy interfejsu SPI, co wyjaśniono w **tabeli 1**.

W protokole SPI przewidziano ramkę zawierającą od 5 do nawet 32 bitów (standardowo 8). Nie wykorzystuje się bitów parzystości. Z oczywistych powodów nie ma też bitów startu i stopu. W większości praktycznych zastosowań bity są wysyłane w kolejności od najstarszego do najmłodszego. Transmisja synchroniczna nie wymaga stosowania znormalizowanych szybkości przesyłania danych, jest ograniczona głównie możliwościami fizycznymi układów elektronicznych i mediów przesyłających sygnały. Częstotliwość przebiegu SCLK może dochodzić nawet do 100 MHz.

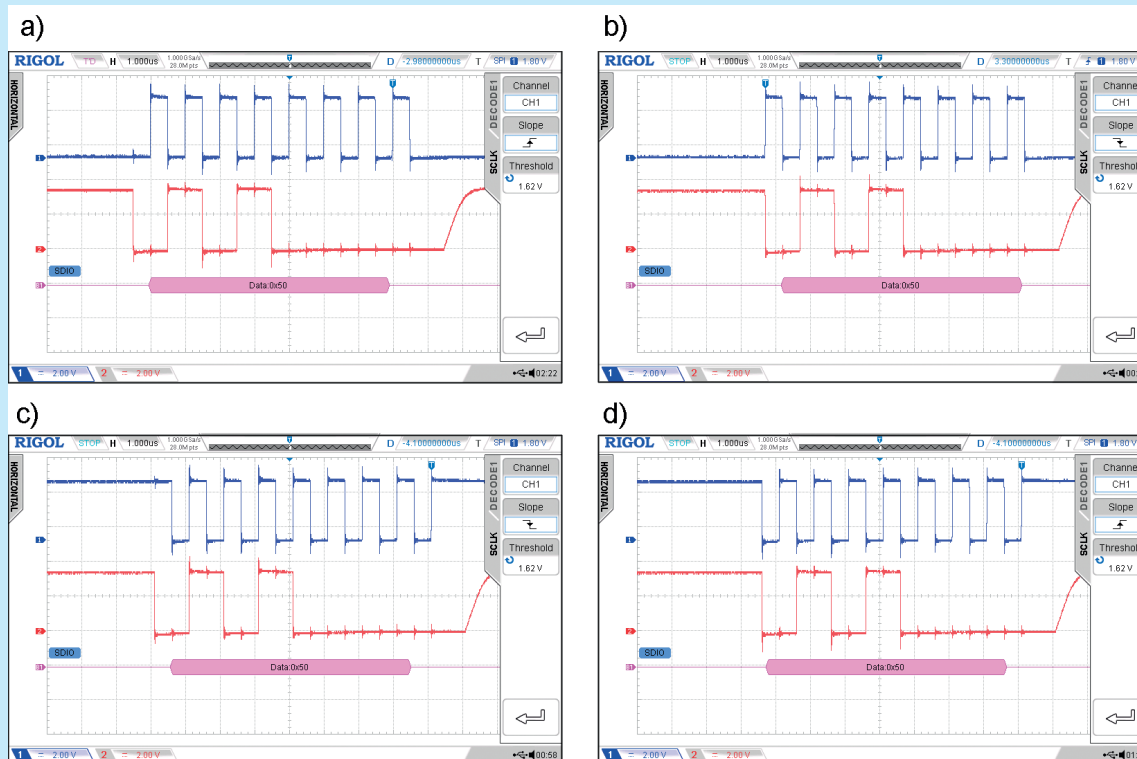
Analiza protokołu SPI

Podstawowym problemem związanym z analizą protokołu SPI za pomocą oscyloskopu Rigol DS2202 jest ograniczona do 2 liczba kanałów pomiarowych. Z tego względu zrezygnowano z badania linii SS. Analityzator nie rozróżnia też linii komunikacyjnych MOSI

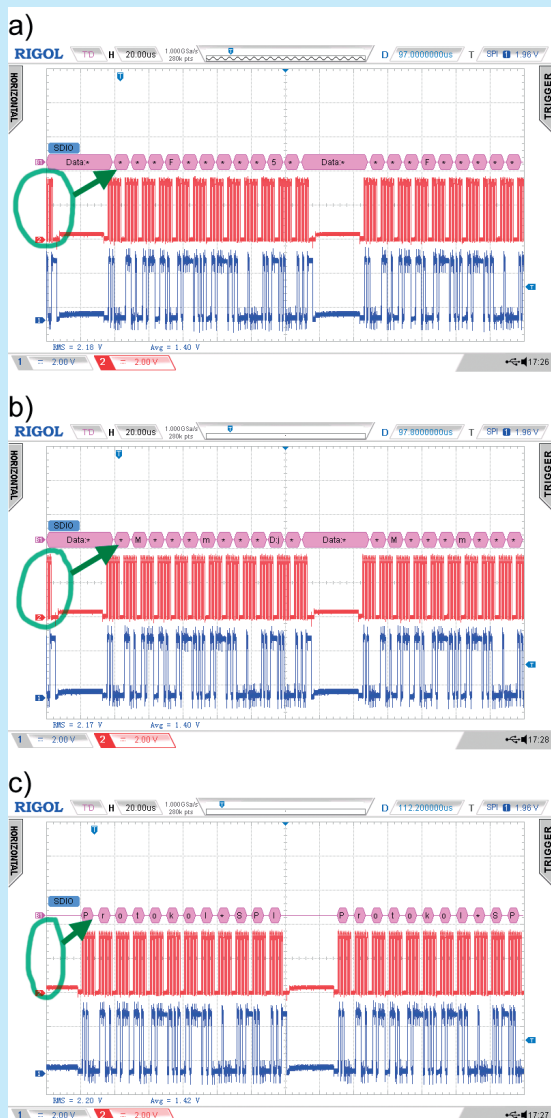
i MISO, zamiast nich linię komunikacyjną nazwano ogólnie SDA (Serial Data). Menu konfigurowania analizatora protokołu SPI przedstawiono na **rysunku 23**. Kanały oscyloskopu nie są na sztywno przypisane do poszczególnych sygnałów. Po rozwinięciu opcji „SCLK” i „SDA” można dokonać odpowiedniego skojarzenia kanałów z sygnałami interfejsu. W tym podmenu definiowane są też dwa parametry decydujące o trybie pracy analizatora, przy czym nie mają one bezpośredniego związku z parametrami interfejsu CPOL i CPHA. W opcji „SCLK” widzimy parametr *Slope*, który wybiera zbocze sygnału zegarowego (narastające lub opadające), na którym następuje odczyt danej z linii SDA (czyli MOSI lub MISO) – **rysunek 23a**. Analityzator nie musi mieć zdefiniowanej fazy CPHA, określana jest za to polaryzacja sygnału. Parametr ten ustawiamy w opcji „SDA” → *Polarity* (*Low* lub *High*). „Polarity”=Low oznacza, że niski poziom napięcia odpowiada logicznemu zeru, natomiast wysoki poziom logicznej jedynce. Odwrotnie jest, gdy „Polarity”=High. Wtedy niski poziom napięcia odpowiada logicznej jedynce, a wysoki poziom logiczemu zeru. Ostatecznie będą nas właściwie interesować tylko przypadki, w których brany jest pod uwagę poziom



Rysunek 23. Menu konfigurowania analizatora protokołu SPI



Rysunek 24. Możliwe kombinacje konfiguracji protokołu SPI w oscyloskopie DS2202



Rysunek 25. Wpływ pozycjonowania oscylogramu w osi czasu na poprawność działania analizatora protokołów

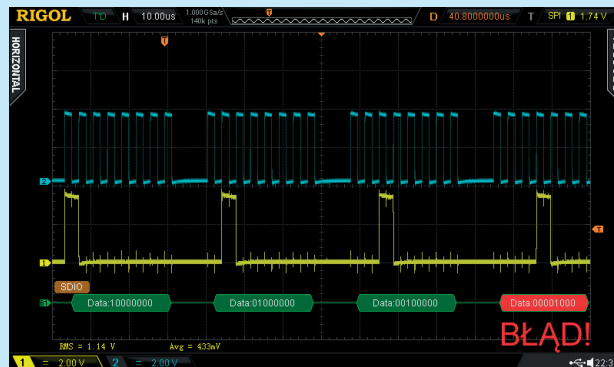
linii SCLK w czasie nieaktywności interfejsu i zbocze sygnału SCLK, na którym jest czytany znak w analizatorze protokołu. Wszystkie kombinacje przedstawiono na **rysunku 24**.

Przed przystąpieniem do pomiarów należy określić pozostałe parametry interfejsu SPI, a więc liczbę bitów danej („Data bits”) oraz kolejność wysyłania bitów („Endian”). Zgodnie z ogólną zasadą, jeśli opcji „Endian” nadano wartość MSB, oznacza to, że bity są wysyłane od najstarszego do najmłodszego, natomiast dla „Endian”=LSB bity są wysyłane od najmłodszego do najstarszego. W większości przypadków będziemy wybierać opcję MSB.

Ostatnim parametrem jest „Format”. Definiuje on sposób interpretowania zdekodowanych danych. Może to być zapis heksadecymalny, dziesiętny, binarny lub w postaci znaków ASCII.

Wyzwalanie SPI

Analizator mamy już prawie gotowy do pracy, brakuje tylko warunku wyzwolenia. Badając protokół SPI sięgamy naturalnie po opcje dostępne dla takiego pomiaru. W menu „Triger” wybieramy opcję *SPI*, a w niej



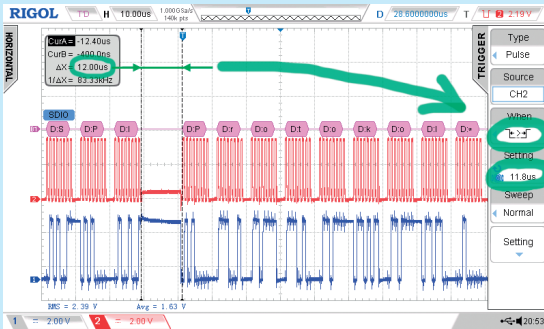
Rysunek 26. Błąd odczytu danej niemieszczącej się na ekranie

jedynym zdarzeniem, z którego możemy skorzystać jest wyzwolenie na określonej danej przesyłanej interfejsem. Definiowanie danej polega na odpowiednim ustawianiu kolejnych jej bitów, co jest metodą skuteczną, ale niezbyt wygodną. Należy zwrócić uwagę na to, że bity są numerowane w kolejności odwrotnej, do jakiej jesteśmy przyzwyczajeni – najstarszy jest bit numer 0, najmłodszy bit dla danej 8-bitowej ma numer 7. Konieczne jest jeszcze powtarzne, niezależne od ustawień protokołu, ale zgodne z nim, przypisanie linii interfejsu SPI (SCLK i SDA) do kanałów oscyloskopu. Dla uzyskania stabilnego oscylogramu, w większości przypadków konieczne będzie przełączenie układu akwizycji w tryb Normal.

Po wykonaniu opisanych czynności spodziewamy się wyzwolenia oscyloskopu na zdefiniowanej danej i wyświetlenia stabilnego oscylogramu. Pojawiają się jednak dwie wątpliwości. Pierwsza dotyczy sytuacji, gdy w przesyłanym komunikacie występuje więcej niż jeden znak podany jako wzorzec dla układu wyzwiania. Istnieje niebezpieczeństwo wyzwiania na każdym z nich, i dlatego należy raczej szukać innej danej lub zupełnie innego warunku wyzwiania. Druga wątpliwość jest związana z ograniczoną liczbą kanałów. Zauważmy, że odbiornik analizatora protokołu SPI zaczyna dekodować daną po wystąpieniu 8 impulsów zegarowych (zakładamy, że z takimi ramkami mamy do czynienia). Prawdziwy odbiornik interfejsu SPI rozpoczyna zliczanie impulsów zegarowych zawsze po uaktywnieniu linii SS, której jednak analizator oscyloskopu w ogóle nie bada. Jak zatem pracuje odbiornik analizatora? Informacji takiej nie znajdziemy w instrukcji oscyloskopu, ale stosunkowo łatwo można zauważyć, że jako pierwszy jest liczony pierwszy impuls zapisany w rekordzie akwizycji. Bardzo prawdopodobne jest, że będzie to środkowy bit którejś z danych, a to oznacza, że kolejne dane będą interpretowane nieprawidłowo. Przykłady takie przedstawiono na **rysunku 25**. Jak widać, dopiero po odpowiednim przesunięciu oscylogramu w osi czasu, komunikat jest dekodowany prawidłowo (**rysunek 25c**). Przekłamanu ulega też dana nie mieszcząca się w rekordzie. Wystarczy, że poza ekranem znajdzie się choćby jeden impuls zegarowy, aby dana została zdekodowana błędnie. Jest wówczas wyświetlana na czerwonym tle (**rysunek 26**).

Ustalenie odpowiedniego warunku wyzwiania jest niezbędne do poprawnej pracy analizatora protokołów. Opcje wyzwiania przypisane do danego proto-

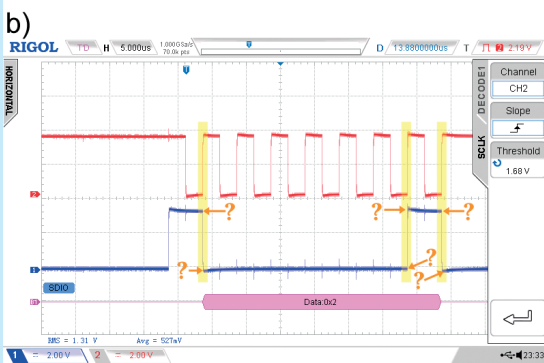
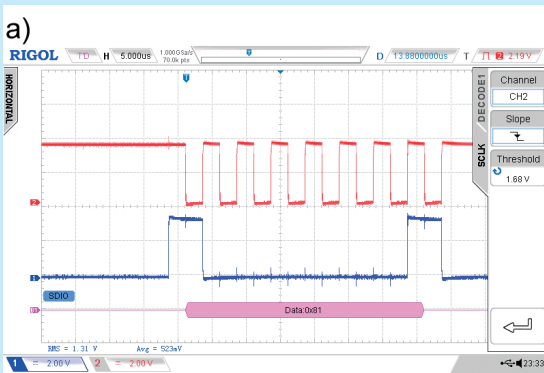
kołu nie zawsze są najlepszym wyborem, często równie dobre okazuje się po prostu wyzwalanie zboczem lub impulsem. Na przykład, jeśli dane są przesyłane w paczkach z pewnymi minimalnymi odstępami, bardzo dobre wyniki daje wyzwalanie impulsem o zadanej szerokości, tak jak to robiliśmy badając interfejs RS-232. Przykład takiego wyzwalania przedstawiono na **rysunku 27**.



Rysunek 27. Wyzwalanie analizatora protokołu SPI impulsem o określonej szerokości



Rysunek 28. Błąd interpretacji danych przy nieprawidłowym ustawieniu parametru „Endian”



Rysunek 29. Interpretacja danych dla różnych nastaw parametru „Slope” a) ustawienie prawidłowe, b) ustawienie błędne

Skutki błędnych nastaw

Ustawienie parametrów protokołu niezgodnych z parametrami badanego urządzenia powoduje oczywiście błędną interpretację odczytywanych przez analizator danych. Tak, jak w przypadku protokołu RS-232, nie zawsze możemy być świadomi takiego stanu. O nieprawidłowym wybraniu parametru „Endian” zorientujemy się natychmiast, jeśli będą transmitowane znaki ASCII – przesyłane informacje będą zupełnie nieczytelne. Jeśli jednak dane mają interpretację liczbową, natychmiastowe wykrycie błędu nie będzie oczywiste (**rysunek 28**).

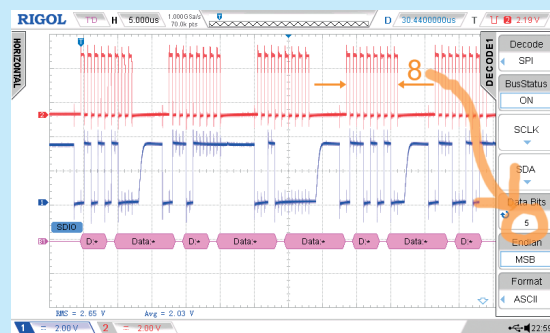
Błąd dekodowania nie zostanie również wyświetlony, gdy zostanie wybrane złe zbocze impulsu zegarowego. Na **rysunku 29** przedstawiono oscylogram uzyskany podczas badania interfejsu SPI pracującego w trybie 2 (CPOL=1, CPHA=0). Podczas nieaktywności interfejsu linia SCLK jest utrzymywana na poziomie logicznej „1”, zmiany poziomów linii danych następują na narastającym zboczu impulsów zegarowych, natomiast odczyt danych na zboczu opadającym. Opcja „Slope” powinna więc być ustawiona na zbocze opadające. Wybranie zbocza narastającego, tak jak na rysunku 29b, powoduje, że może być odczytany zarówno poziom bezpośrednio przed impulsem, jak też za nim. Decydują o tym wręcz warunki propagacji sygnału w danym urządzeniu. Przy odrobinie szczęścia może zdarzyć się, że odczyt będzie poprawny, nie mniej, mamy do czynienia z błędną konfiguracją analizatora.

Pozostała jeszcze reakcja na nieprawidłowo ustawioną długość ramki. Jeśli kolejne znaki są nadawane z małymi przerwami, wychwycenie tego błędu nie będzie trudne. Szybko zauważymy, że długość danych nie odpowiada paczkom impulsów zegarowych, poza tym dane będą dekodowane błędnie (**rysunek 30**). Gorziej, jeśli transmisja odbywa się bez przerw między kolejnymi znakami, co jest w interfejsie SPI możliwe. Jedynym objawem błędu jest wówczas nieprawidłowa interpretacja danych. Oscyloskop nie sygnalizuje jednak żadnego błędu w takim przypadku.

Omówiliśmy zagadnienia związane z analizą protokołu SPI. Należy dodać, że część uwag dotyczących RS-232 jest wspólna dla wszystkich interfejsów obsługiwanych przez analizator protokołów oscyloskopu Rigol DS2202. Zdekodowane dane nadal można przeglądać w oknie Zoom oraz w tabeli wyników.

W następnym odcinku omówimy badanie interfejsu I²C.

Jarosław Doliński, EP



Rysunek 30. Błąd interpretacji danych wynikający ze złego podania długości ramki