

HSIC czy USB?

Interfejs HSIC zyskuje w ostatnim czasie na popularności. Od USB różni się przede wszystkim warstwą fizyczną zachowując cały stos programowy i przesyłając dane z prędkością 480 Mbit/s. Służy do łączenia ze sobą układów scalonych lub podzespołów w obrębie obudowy urządzenia. Ze względu na liczne zalety interfejsu HSIC liczba jego zastosowań stale rośnie i można się spodziewać, że w najbliższym czasie ten trend będzie się nasilał.

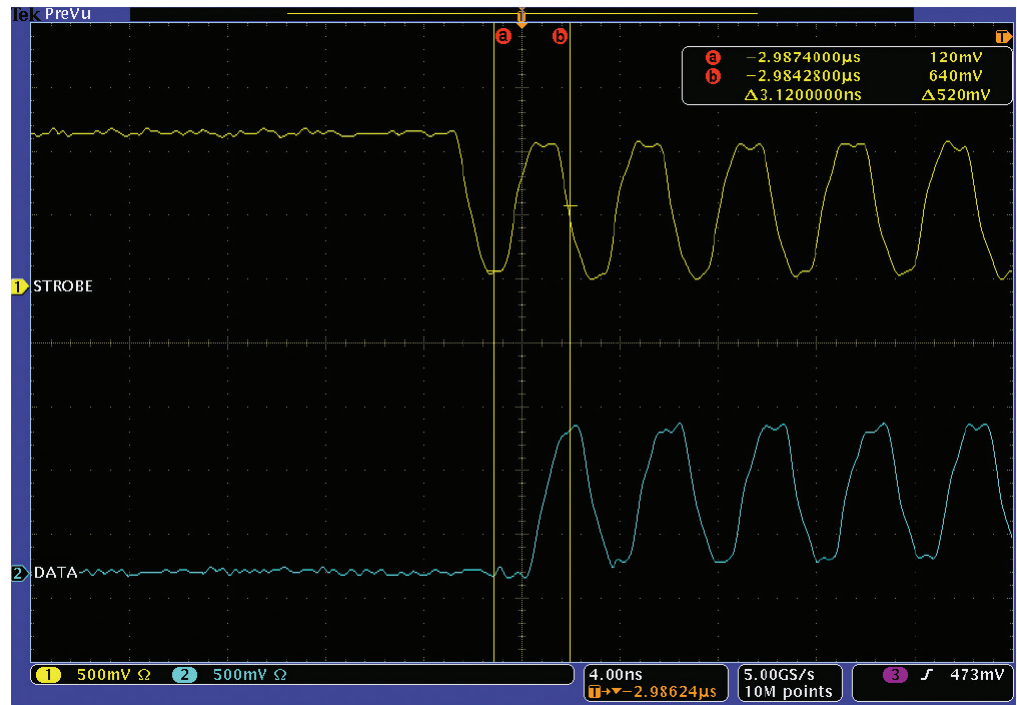
Skrót HSIC oznacza *High-Speed Inter-Chip* i odnosi się do coraz bardziej popularnego, dwusygnałowego, synchronicznego interfejsu komunikacyjnego. Pod względem logicznym jest on zgodny z USB, dzięki czemu może łatwo zastępować ten popularny standard. Różni się natomiast w warstwie fizycznej – nie obsługuje protokołu chirp do rozpraszania widma, źródło przesyła dane synchronicznie oraz nie obsługuje mechanizmów dołączania i odłączania urządzeń w trakcie pracy. Nie jest to bynajmniej wadą, gdyż z założenia służy do połączeń wewnątrz urządzeń, a więc takich, które są na stałe. Pracuje z napięciem 1,2 V, dzięki czemu dobrze sprawdza się w aplikacjach zasilanych bateryjnie oraz wymagających niskiego zużycia mocy. Maksymalna odległość długość połączenia pomiędzy elementami sprzęgniętymi za pomocą interfejsu HSIC wynosi 10 cm.

To co odróżnia HSIC od USB w warstwie fizycznej, to przede wszystkim fakt, że wszelkie dane przesyłane są za pomocą jednej linii. Druga linia służy tylko do przesyłania sygnału zegarowego, który informuje o tym, kiedy należy próbkować sygnał na pierwszej linii. Ponieważ HSIC korzysta z techniki DDR (*Double Data Rate*), dane są próbkowane zarówno na opadającym, jak i rosnącym zboczach sygnału zegarowego. Biorąc pod uwagę fakt, że sygnał zegarowy ma częstotliwość 240 MHz, sumaryczna przepustowość interfejsu wynosi 480 Mb/s.

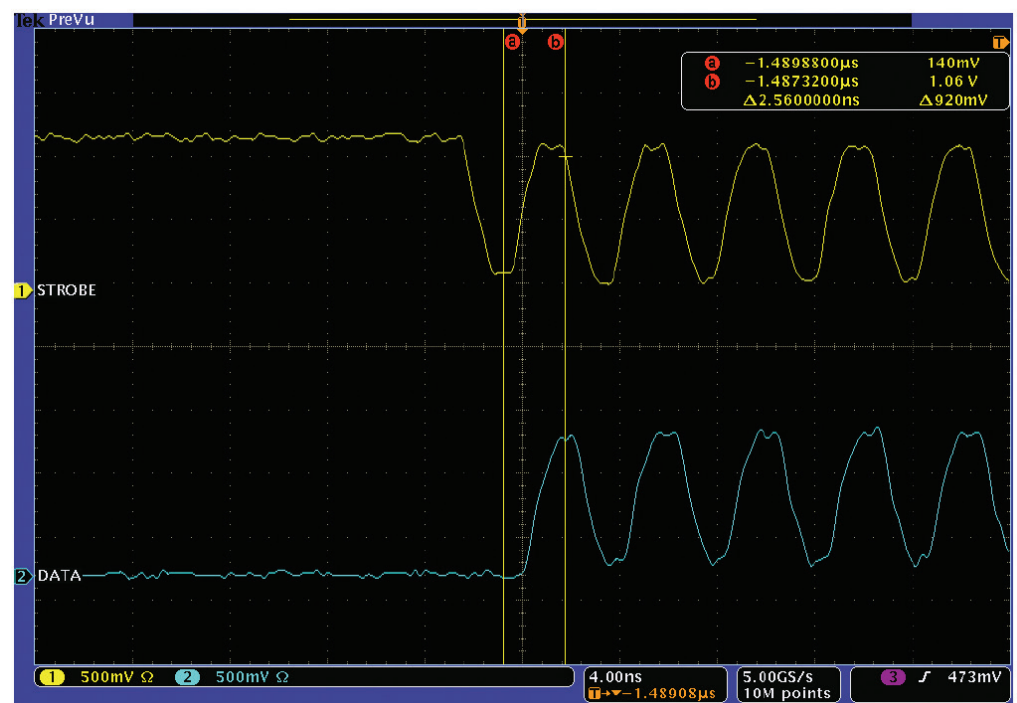
Zalety względem USB

W porównaniu z USB interfejs HSIC ma szereg zalet. Po pierwsze, jest w pełni cyfrowy, co ozna-

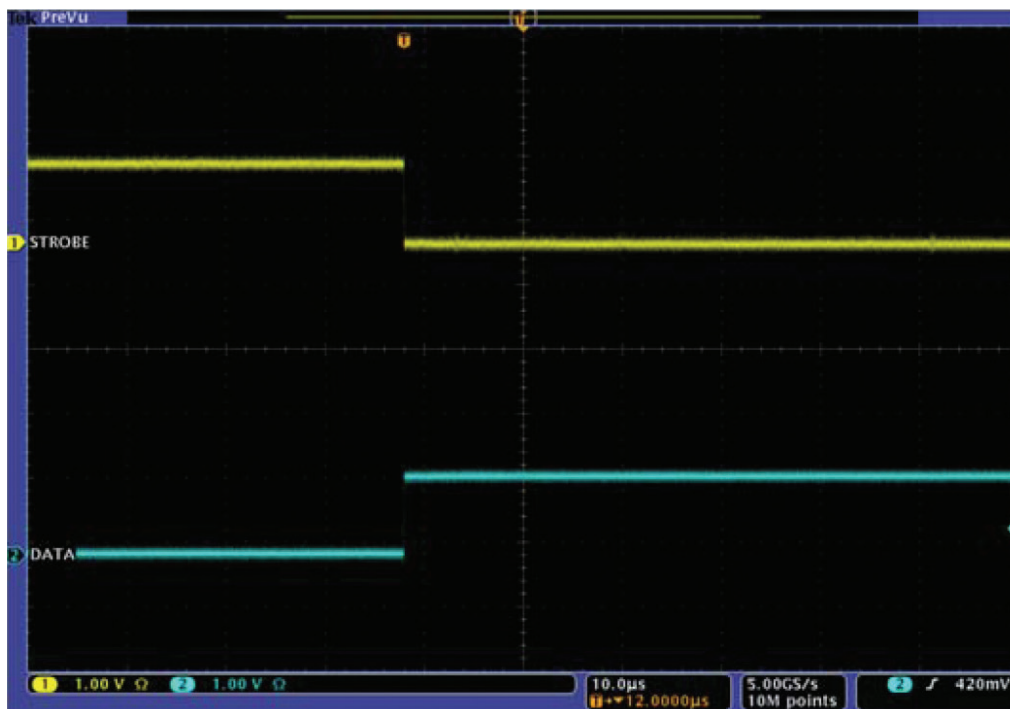
cza, że nie ma potrzeby projektowania obwodów analogowych i w konsekwencji nie zajmują one cennej powierzchni układu scalonego. Pozwala to nie tylko na zmniejszenie jego wielkości, ale również obniżenie ceny. Maleją też wymagania odnośnie do rdzenia dzięki uproszczeniu logiki potrzebnej do obsługi protokołu.



Rysunek 1. Sygnały HSIC na ścieżkach o identycznej długości



Rysunek 2. Sygnały HSIC na ścieżkach różniących się długością o 10 cm



Rysunek 3. Sygnały HSIC na liniach w trakcie łączenia się urządzeń

Pewne zalety są związane też z obniżeniem zużycia mocy. Co prawda, w trakcie normalnej pracy interfejsu różnica jest znikoma, bo wynika z braku obwodów analogowych, które w USB pobierają niewielką moc. Większą różnicę widać dopiero po przejściu w tryb wstrzymania, w którym linie sygnałowe nie pobierają żadnego prądu. Dla porównania, w analogicznym trybie pracy interfejsu USB, linia D+ pobiera przynajmniej 200 μ A prądu, który przepływa przez rezystor podciągający o wartości 1,5 k Ω .

Próbkowanie danych

W interfejsie USB każdy pakiet danych rozpoczyna się od sekwencji synchronizującej, która ułatwia odbiorcy dostosowanie swojego sygnału zegarowego do fazy przychodzących danych. Po tym procesie, sygnał różnicowy odczytywany z linii D+ i D- jest próbkowany zgodnie z wyliczonym przesunięciem w fazie.

W interfejsie HSIC nie ma potrzeby przesyłania pakietu synchronizacyjnego, gdyż moment próbkowania danych jest określany przez niezależną linię zegarową. Specyfikacja elektryczna HSIC określa, że maksymalna niedokładność sygnału zegarowego może wynosić 15 ps. Aby to było możliwe, zdecydowano że długość połączeń realizowanych z użyciem HSIC nie może być większa niż 10 cm, a ścieżka danych i zegarowa mieć tę samą długość i powinny być zakończone terminatorami 50-omowymi.

Fakt, że ograniczenia te pozwalają utrzymać integralność transmisji, widać na rysunkach. Na **rysunku 1** pokazano przebieg sygnału zegarowego (STROBE) i sygnału danych (DATA), gdy obie ścieżki sygnałowe są tej samej długości. Na **rysunku 2** zaprezen-

towano natomiast, co się dzieje, gdy ścieżka danych jest o 10 cm dłuższa, niż ścieżka zegarowa. Skutkuje to przesunięciem zegara o około pół nanosekundy względem sygnału danych, co jest niezgodne ze specyfikacją HSIC i uniemożliwia poprawną transmisję danych z użyciem tego interfejsu.

Sposób terminowania linii sygnałowych w HSIC powoduje pewne trudności z próbkowaniem i analizą przesyłanych sygnałów. O ile typowe linie USB mogą być łatwo monitorowane za pomocą sond oscyloskopowych umieszczonych przy komponencie nadającym lub odbierającym dane, o tyle linie HSIC są bardziej wrażliwe. Należy się liczyć ze zjawiskami falowymi, które zachodzą w momencie dołączania do nich sond. W efekcie zaleca się umieszczanie sond tuż za elementem odbierającym dane. Ponieważ transmisja HSIC jest dwukierunkowa, oznacza to konieczność przełączania sond w zależności od tego, który z komunikujących się układów będzie nadawcą danych. Uwzględniając nomenklaturę stosowaną odnośnie urządzeń USB, oznacza to, że jeśli nadawcą ma *host*, sondy należy dołączyć w pobliżu urządzenia pracującego w trybie *device*. Jeśli nadawcą ma urządzenie komunikujące się w trybie *device*, sonda powinna znaleźć się przy *hoście*. W przeciwnym wypadku, odbierany sygnał będzie zniekształcony, co wynika z powstających przy sondzie odbić. Włączenie sond w środku linii też nie daje tak dobrych rezultatów, jak w przypadku umieszczenia sond przy odbiorniku.

Wykonywanie połączenia

Pozostaje pytanie, jak skorzystać z dobrodziejstw HSIC i jak się posługiwać tym in-

terfejsem. Korzystne jest, że kolejność dołączenia do zasilania urządzeń pracujących w trybie *hosta* i w trybie *device* nie ma znaczenia. Aby upewnić się, że nie zostanie wykryte nieprawdziwe połączenie, linie sygnałów i danych powinny zostać podciągnięte do stanu wysokiego lub niskiego, by nie występował na nich stan nieokreślony, taki jak zaprezentowany na **rysunku 3**.

Nawiązanie połączenia też jest bardzo proste, gdyż w przeciwieństwie do USB, nie wymaga negocjacji szybkości przesyłu danych. W efekcie, obwody logiczne konieczne do ustanowienia połączenia mogą być proste i nie zajmują dużej powierzchni układu scalonego.

W przypadku USB konieczne jest też zapewnienie obsługi nagłego odłączenia urządzenia

pracującego w trybie „device”. Odbywa się to poprzez monitorowanie napięć na liniach sygnałowych, które jeśli przekroczą określone wartości, są interpretowane jako objaw odłączenia urządzenia. W przypadku HSIC odłączenie urządzenia nie jest obsługiwane, gdyż zakłada się, że połączenia wykonane są na stałe w postaci ścieżek na płycie drukowanej, przylutowanych do wyprowadzeń komunikujących się układów. Mimo to konieczne jest obsłużenie sytuacji, w której urządzenia pracujące w trybie *device* wydaje się być odłączone. Sytuacja taka może wystąpić, ponieważ *host*, niezależnie czy jest bezczynny, czy jest w stanie uśpienia, utrzymuje identyczny stan linii sygnałowych. Może więc zdarzyć się, że urządzenie podrzędne zakłada, że zostało uśpione przez *hosta*, a *host* oczekuje na sygnał od urządzenia podrzędnego. Wtedy to należy zapewnić, że połączenie nie zostanie utracone na stałe. Realizuje się to najczęściej za pomocą odpowiedniego oprogramowania stosu komunikacyjnego. Alternatywnie, *host* może próbować zresetować hub HSIC, co spowoduje ponowne wykrywanie podłączonych urządzeń i ponowne nawiązanie połączenia.

Podsumowanie

Zastosowanie HSIC zamiast USB do realizacji pomiędzy połączeniami wewnątrz urządzeń elektronicznych jest korzystnym rozwiązaniem. Ponieważ HSIC różni się od USB tylko w warstwie fizycznej, migracja do nowszego standardu jest bardzo prosta, gdyż praktycznie nie wymaga wprowadzania zmian do używanych stosów protokołów.

Andrew Rogers,
Microchip