

SOC-System on Chip

CISC (complex instruction set computer)	RISC (reduced instruction set computer)
złożone instrukcje	proste instrukcje
specjalizowane rejestry	bez specjalnych rejestrów
mała liczba ogólnych rejestrów nauczyć się korzystać z tych specjalnych	w cholerę ogólnych rejestrów
przedstawiciel: SUPREME8051	przedstawiciel: virgin AVR

Architektury:

Architektura von Neumanna	Architektura Harvardzka
pojedyncze magistrale danych i adresowa zarówno do programu jak i danych	różne magistrale danych i adresowa
słowo danych i rozkaz tej samej długości	długości słowa mogą być różne
dostęp do danych i programu w różnych cyklach (von Neumann bottleneck)	dostęp do danych i programu w 1 cyklu
pamięć danych w tym samym miejscu co pamięć programu	pamięć programu w innym miejscu niż pamięć danych
zastosowanie w procesorach Intelu typu 8051	zastosowanie w układach typu AVR

UART,USART

UART	USART
tylko asynchroniczny	i synchroniczny i asynchroniczny
jak chcesz coś nadać albo odebrać to się najpierw musisz ustawić w nadajniku i odbiorniku prędkość i to taką samą	możesz dać tryb synchroniczny i się nie przejmować
nie ma sygnału zegarowego	może mieć sygnał zegarowy
może mieć 1,1.5,2 bity stopu	może mieć 1,1.5,2 bity stopu
chory degenerat który zawsze się spóźnia	można go postawić do pionu

Czy IP-Core to moduł obsługi sieci występujący w niektórych mikrokontrolerach? **Nie**

Czy moduł PVD w STM32 pozwala na pomiar wartości napięcia zasilania? **Nie.**

Czy wszystkie rejestry w strukturze ARM są mapowane w pamięci? **Nie**

Czy w ARM reset ma wyższy priorytet niż NMI? **Tak**

Czy adresy rejestrów portów GPIO w ARM mieszczą się w obszarze pamięci 0x40000000 - 0x5FFFFFFF? **Tak**

Czy w przypadku układów wyposażonych w interfejs TWI programista ma pełną swobodę w przyporządkowaniu adresów poszczególnych układów peryferyjnych podłączonych do magistrali? **Nie**

Czy w przypadku gdy na magistrali TWI odbywa się sekwencyjny odczyt większej liczby bajtów z urządzenia SLAVE do MASTER to MASTER jest odpowiedzialny za generowanie sygnału ACK? **Tak**

Czy w STM32 jest możliwe aktywowanie tylko wybranych układów peryferyjnych w celu ograniczenia poboru energii? **Tak**

Czy układ STM32 w przypadku awarii generatora HSE może go zastąpić LSE? **Nie. To są dwa różne generatory. HSE jest tym głównym taktującym procesor. LSE jest odpowiedzialny za zegar RTC. LSE można zastąpić HSE, ale nie będzie podtrzymywania baterijnego (np. do RTC).**

Czy w STM32 każdy pin GPIO potencjalnie może służyć jako wejście zewnętrznego przerwania? **Tak**

Czy w przypadku określonych układów wyposażonych w interfejs TWI istnieje możliwość modyfikacji w niewielkim zakresie ich adresów, które następnie są wykorzystywane przy odwoływaniu się do tych układów? **Tak**
Każdemu z urządzeń przypisuje się indywidualny adres.

Czy w listach rozkazów proc. RISC występują (typowo) rozkazy przesyłu bloków danych? **Nie**

Czy wszystkie rejestry w strukturze ARM są mapowane w pamięci? **Nie**

Czy adresy rejestrów portów GPIO w ARM mieszczą się w obszarze pamięci 0x60000000 - 0x7FFFFFFF? **Nie**

Czy zwarcie linii układów TWI jest możliwe dzięki temu, że wyjścia te są trójstanowe? **Tak**

Czy na magistrali TWI sygnał zegarowy występuje zawsze, niezależnie od stanów, w jakich znajdują się urządzenia Master i Slave? **Tak**

Czy w procesorze STM32 jest możliwe aktywowanie tylko wybranych układów peryferyjnych w celu ograniczenia poboru energii? **Tak**

Czy zwarcie linii układów TWI jest możliwe dzięki temu, że wyjścia te są typu OD? **Nie**

Czy w listach rozkazów proc. RISC występują (typowo) rozkazy przesyłu bloków danych? **Tak**

Czy w procesorach ARM wszystkim przerwaniom/wyjątkom można przyporządkować "własny" priorytet? - **NIE, niektóre są stałe (NMI, Reset, HardFault)**

Jednoczesne uaktywnianie się sygnałów nie /IORQ i nie /M1 na wejściach Z-80PIO informuje, że procesor oczekuje na wystawienie na magistrali wektora obsługi przerwania. **TAK**

Czy w przypadku łączenia układów peryferyjnych rodziny Z-80 (PIO, CTC SIO i in.) łańcuch "daisy chain" ich wyjścia INT powinny być ze sobą zwarte i podłączone razem do wejścia INT procesora Z-80. **TAK**

W 8255 ustawianie i zerowanie bitów wyjściowych portu PC przy pracy w trybie 0 zachodzi poprzez wpisanie odpowiedniego słowa sterującego pod adres rejestru sterującego. **NIE**

Czy w układzie Z-80PIO każdy port, który może prowadzić transmisję z potwierdzeniem posiada swoją własną linię INT? **NIE**

W trakcie inicjalizacji portu szeregowego można ustawić port tak aby generował 1.5 bitu startu. **TAK**

Czy procesor Z-80 jest włączony wraz z innymi układami rodziny Z-80 w łańcuch priorytetu przerw (daisy chain) stanowiąc jednostkę nadrzędną w tym łańcuchu? **NIE**

W systemie komputerowym opartym na mikroprocesorze Z-80 chcąc zwiększyć obciążalność magistrali procesora należy obowiązkowo zastosować rejestry, które będą przechowywały zawartość magistrali. **TAK**

Jeżeli w 8255 port A i B pracują w trybie wyjściowym z potwierdzeniem to port C można wykorzystać jako prosty 8-bitowy port wejściowy lub wyjściowy. **TAK**

Priorytet przerw w łańcuchu daisy chain jest ustalany przez fizyczne (w sensie połączeń elektrycznych) położenie każdego z układów (np. rodziny Z-80) względem innych. **TAK**

Jeżeli w trakcie inicjalizacji ustawiono port szeregowy na generowanie 2 bitów stopu, to w trakcie dowolnej transmisji danych zawsze po dwóch bitach stopu wystąpi bit startu? **NIE**

Czy z poziomu programu działającego na Z-80 istnieje możliwość zmiany priorytetów przerw pomiędzy układami z rodziny Z-80 połączonymi w łańcuch daisy-chain? **prawdopodobnie tak**

1. Rozwiń skrót MAC (w dziedzinie DSP) **M**ultiply and **A**cCumulate
2. W której z podstawowych architektur systemów komputerowych można łatwo doprowadzić do wykonania programu w obszarze pamięci zarezerwowanej dla danych? **Von Neumann'a**
3. Co w mikrokontrolerach ARM(STM32) może pełnić rolę pamięci EEPROM znanej z AVR?

Backup register (nie może być SRAM, bo jest to pamięć ulotna.) Obszar podtrzymywany bateryjnie.

4. Na czym polega mechanizm bit-banding występujący w ARM(STM32)?

Można traktować dany pin jak zmienną i np. przekazywać pin jako parametr funkcji, bez używania do tego rejestrów. Poszczególne bity w rejestrach można zaadresować jako całe rejestry o innych adresach. Wpisanie do/zczytanie czegoś z takiego rejestru pozwala operować na pojedynczych bitach, na co nie pozwalają same rozkazy ARM. Istotny tak naprawdę jest ostatni bit wirtualnego rejestru - on zdecyduje o stanie modyfikowanego bitu w docelowym, fizycznym rejestrze.

5. Co w ARM oznacza że wyjątek ma status „active and pending”?

Wyjątek jest obsługiwany przez procesor i istnieje oczekujący wyjątek z tego samego źródła.

6. Do magistrali SPI zostały podłączone 4 urządzenia Slave w jaki sposób Master dokonuje wyboru, z którym będzie się komunikował?

Każdy z Slave'ów ma osobną linię SS - slave select. Po podaniu stanu aktywnego (zwykle niski) na tą linię aktywujemy podpięty do niej układ. Można jednocześnie aktywować parę układów i np. zapisywać do nich dane.

7. Do czego służą kondensatory o wart. np. 10nF podpinane równolegle pomiędzy liniami zasilania i masami układów cyfrowych?

Filtruje napięcie zasilające by zabezpieczyć układ przed zakłóceniami np.(szpilki zapaści) Są to kondensatory odsprężające, niwelują zakłócenia wynikające z impulsowego charakteru poboru prądu przez układy cyfrowe.

8. Układ STM32 może „samodzielnie” zaprogramować własną pamięć FLASH(pamięć programu). W jaki sposób się to odbywa, skąd pobierane są dane?

Programowanie „samodzielne” jest dokonywane z pomocą bootloadera. Jest to mały program który uruchamia się zawsze jako pierwszy i może zacząć komunikować się z innym układem (komputerem) np.przez USB i odpytać go o nowy kod. Po czym ściąga ten kod i wrzuca do pamięci flash, po czym wykonuje skok do tego nowo zapisanego kodu. (Bootloader wrzuca się na sam początek Flasha. Dlatego wystartuje jako pierwszy i zaprogramuje co trzeba. Nic nie trzeba skonfigurować, tylko układ musi być tak ustawiony żeby uruchamiał się z Flash - realizacja dwoma pinami których stan (0/1) determinuje start z Flash / RAM.)

9. Czy w STM32 wejścia/wyjścia wewnętrznych układów peryferyjnych są jednoznacznie przypisane do określonych pinów układu ? T/N

Nie. Każde fizyczne wyjście (nóżka układu) może pełnić różne funkcje, zmieniane programowo.

10. Na czym polega zasadnicza funkcjonalność „window watchdog” w STM32?

Window watchdog działa jak zwykły watchdog - jest to timer który programowo resetuje układ jeżeli nie zostanie „odświeżony”. Window watchdog dodatkowo wymaga żeby odświeżenie odbyło się w określonym oknie czasowym- nie za szybko, nie za późno.

11. Jaka jest różnica między interrupt i event?

Każde przerwanie jest eventem, ale nie każdy event jest przerwaniem. Można powiedzieć, że interrupt jest formą natychmiastowego powiadomienia procesora o ważnym zdarzeniu, podczas gdy event jest formą powiadomienia, które jest dodawane do kolejki i jest obsługiwane w odpowiedniej kolejności.

12. Rozwiń skrót GPIO –

General Purpose Input/Output

Wejście/Wyjście Ogólnego Zastosowania

13. Na czym polega zjawisko „von Neumann bottleneck” i czy jest to zjawisko pozytywne czy negatywne (uzasadnij)

Dostęp do kodu programu i danych następuje w odrębnych cyklach, ponieważ CPU jest bierny podczas dostępu do pamięci - zjawisko negatywne, ponieważ nakłada ograniczenie na przepustowość

14. Jak w ARM (STM32) przechować dane pomiędzy wyłączeniami zasilania układu?

* Pamięć flash emulowana jako EEPROM

* Rejestry backupowe (20 bajtów) podtrzymywane baterijką

15. Co kryje się pod pojęciem "Thumb" używanym w przypadku procesorów ARM?

Zestaw 16-bitowych rozkazów zaprojektowanych w celu oszczędności z punktu widzenia pamięci

16. Czy moduł PVD w STM32 pozwala na pomiar wartości napięcia zasilania?

Nie, ponieważ PVD jest komparatorem napięcia

17. Jaki mechanizm udostępniają producenci w niektórych układach z mag. TWI w celu umożliwienia zróżnicowania adresów układów tego samego typu podłączonych do tej samej magistrali?

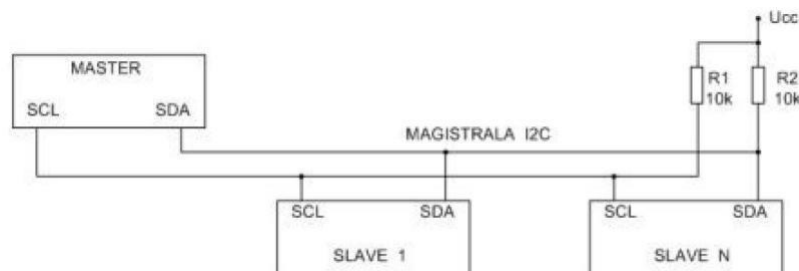
Wyrowadzenie pinów adresowych do adresacji manualnej

18. Opisz, w jaki sposób układ Master wybiera układ Slave, z którym chce się komunikować w przypadku interfejsu 1 wire.

Master wysyła impuls reset czyli zwarcie linii danych do masy. Powoduje to zresetowanie podłączonych odbiorników. Następnie każde urządzenie slave wysyła impuls obecności. Przesłanie stanu aktywnego na magistrale powoduje aktywację przerzutnika i odczyt danych.

19. Opisz transmisję I2C.

Urządzenia podłączone do magistrali mogą być skonfigurowane jako MASTER (urządzenie nadrzędne) lub SLAVE (urządzenie podrzędne). W danej sesji łączności tylko jeden mikrokontroler podłączony do magistrali może być skonfigurowany jako MASTER, chociaż każde z urządzeń może pracować zarówno jako nadajnik jak i odbiornik. Mikrokontroler MASTER generuje przebieg zegarowy SCL taktujący transmisję oraz inicjuje łączność z wybranym urządzeniem SLAVE podając programowo jego adres. Wybrany SLAVE potwierdza odbiór bajtu adresowego, pozostałe urządzenia SLAVE pozostawiają linię SDA w stanie wysokiej impedancji.



1. Na co należy zwrócić uwagę jeżeli chcemy sprawdzić czy można podłączyć do procesora (np. STM32) 20 x LED, z których każda pobiera 10mA (bez dodatkowych kluczy/driverów/itp.)?

Ważne jest, aby upewnić się, że procesor posiada wystarczającą moc wyjściową, aby podać 20 diod LED jednocześnie, a także że posiada zabezpieczenia przed przeciążeniem i zwarcie, aby uniknąć uszkodzenia sprzętu.

2. Czy w procesorach ARM wszystkim przerwaniom/wyjątkom można przyporządkować "własny" priorytet?

Tak, w procesorach ARM wszystkim przerwaniom/wyjątkom można przypisać "własny" priorytet. Priorytety przerw/wyjątków są ustawiane przez programistę i określają kolejność, w jakiej są one obsługiwane przez procesor. Procesory ARM posiadają rejestry, które pozwalają na ustawianie i zmianę priorytetów dla różnych przerw/wyjątków, dzięki czemu programiści mogą dostosować ich działanie do własnych potrzeb.

Które z fizycznych sygnałów procesora Z-80 są istotne z punktu widzenia dostępu do urządzeń I/O zlokalizowanych we wspólnej lub rozdzielonej przestrzeni adresowej pamięci i układów I/O. Określ, które sygnały wykorzystuje się w każdej z wymienionych sytuacji.

a) Sygnały istotne z punktu widzenia dostępu do urządzeń I/O zlokalizowanych we wspólnej przestrzeni adresowej pamięci i układów I/O to:

- M1 (Memory Request): Ten sygnał jest wysyłany, gdy procesor zaczyna dostęp do pamięci. - IORQ (Input/Output Request): Ten sygnał jest wysyłany, gdy procesor chce dostępu do urządzenia I/O.

- RD (Read): Ten sygnał jest wysyłany, gdy procesor chce odczytać dane z urządzenia I/O.

- WR (Write): Ten sygnał jest wysyłany, gdy procesor chce zapisać dane do urządzenia

b) Sygnały istotne z punktu widzenia dostępu do urządzeń I/O zlokalizowanych w rozdzielonej przestrzeni adresowej pamięci i układów I/O to:

- AEN: aktywuje adresowanie urządzeń I/O za pomocą adresu A15-A8.

- ALE: aktywuje adresowanie urządzeń I/O za pomocą adresu A7-A0.

- MEMRQ / IORQ: określa, czy procesor chce odczytać dane z urządzenia I/O (IORQ) czy z pamięci (MEMRQ).

MEMRD / IOWR: określa, czy procesor chce zapisać dane do urządzenia I/O (IOWR) czy odczytać je z niego (MEMRD).

49. Opisz mechanizm "waitstate" i do czego służy (na przykładzie ATmega128)(2/2pkt):

Jest to stan oczekiwania zanim w mikroprocesorze zostanie wykonana instrukcja, musimy dać wolniejszej pamięci lub jakimś urządzeniom zewnętrznym czas na odpowiedź. Takich waitstatów mogą być jeden lub więcej. Zazwyczaj jak mamy wolniejszą pamięć to stosujemy większą ilość waitstatów żeby dać jej więcej czasu. Same waitstaty są startą ze strony wydajności procesora. Gdy procesor potrzebuje dostępu do pamięci zewnętrznej zaczyna umieszczać adres żądanych danych informacji na szynie adresowej i musi czekać na odpowiedź która może przyjsć dziesiątki cykli.

Mechanizm "waitstate" to stan czekania, ma za zadanie wprowadzenie lekkiego opóźnienia, aż procesor wykona instrukcję. Ma to na celu danie czasu wolniejszym urządzeniom lub pamięci. Dodanie stanów oczekiwania ograniczy szybkość przetwarzania do mniej niż 60 MIPS

50. Do magistrali SPI zostały podłączone 4 urządzenia Slave- napisz w jaki sposób Master dokonuje wyboru, z którym urządzeniem będzie się komunikował? (2/2pkt)

Dla każdego z slave'ów została przewidziana osobna linia slave select (ss). Aktywacja układu na tej linii odbywa się przez podanie na nią stanu aktywnego, w ten sposób możliwe jest jednoczesne aktywowanie pary układów oraz zapisanie do nich danych.

52. Na jakie parametry elektryczne mikrokontrolera należy zwrócić uwagę, jeżeli planujemyysterowanie przy jego pomocy np 16 LED-ów o prądzie $I_F=10\text{mA}$ i napięciu $U_F=2\text{V}$. Uzasadnij odp. Założyć, że rozważana propozycja rozwiązania to podpięcie diod bezpośrednio do wyjść mikrokontrolera (+ rezystory) (1/2pkt)

Należy zwrócić uwagę na rezystory ograniczające prąd, jeżeli planujemy wykorzystać diody bezpośrednio podpięte do wyjść mikrokontrolera należy obliczyć rezystancje oporników z następującego wzoru: $R=(U_{cc}-2*U_{ce}-U_F)/I_F$

53. Opisz, jaka jest zasadnicza różnica w mechanizmie adresowania pamięci statycznych i dynamicznych?

Pamięć statyczna wykorzystuje stały zestaw adresów do przechowywania i odczytywania danych. Oznacza to, że adresy są przypisane do danych, które są tam trwałe. Pozwala to na szybki dostęp, ale ogranicza elastyczność.

Pamięć dynamiczna wykorzystuje zmienny zestaw adresów, który może się zmieniać podczas działania programu. Ta elastyczność oznacza większą wydajność, ponieważ program może być zmaksymalizowany do potrzeb użytkownika. Adresy mogą być również dodawane lub usuwane podczas działania programu.

54. Opisz mechanizm rozszerzenia pamięci RAM w AVR (np. ATmega 128) - jak jest realizowany od strony sprzętowej i opisowo jak wyglądają przebiegi czasowe na istotnych liniach mikrokontrolera. W opisie nie należy wskazać numerów pinów, ale należy

posłużyć się funkcjonalnymi nazwami magistral/pinów (wskazując na ich rozmiar w sensie liczby bitów), które są wykorzystywane przez ten mechanizm.

Mechanizm rozszerzenia pamięci RAM w AVR (np. ATmega 128) jest realizowany za pomocą magistrali adresowej i magistrali danych. Magistrala adresowa składa się z 16-bitowego sygnału adresu, który jest wysyłany do pamięci RAM. Magistrala danych składa się z 8-bitowego sygnału danych, który jest wysyłany do lub z pamięci RAM. Sygnał adresu jest generowany przez mikrokontroler AVR i wysyłany na magistralę adresową. Sygnał ten jest następnie odbierany przez pamięć RAM, która odczytuje lub zapisuje dane do odpowiedniego adresu. Po odebraniu sygnału adresowego, pamięć RAM wysyła sygnał potwierdzenia (RDY) na magistralę danych, informując mikrokontroler AVR o gotowości do pracy. Gdy mikrokontroler AVR otrzyma sygnał potwierdzenia (RDY), może on wysyłać lub odbierać 8-bitowe dane na magistrali danych. Jeśli mikrokontroler chce odczytać dane z pamięci RAM, to musi on najpierw wysłać sygnał żądania odczytu (RQ) na magistrali danych. Gdy pamięć RAM otrzyma ten sygnał, to przesyła 8-bitowe dane na magistrali danych do mikrokontrolera AVR. Jeśli mikrokontroler chce zapisać dane do pamięci RAM, to musi on najpierw wysłać 8-bitowe dane na magistrali danych oraz sygnał żądania zapisu (WQ). Gdy pamięć RAM otrzyma te informacje, to przechowa je pod odpowiednim adresem i wyśle potwierdzenie (RDY) na magistrali danych informując mikrokontroler AVR o poprawnym zapisie/odczycie.

55. Opisz mechanizm “zasilania pasożytniczego” (parasite power) wykorzystywany w przypadku magistrali 1-Wire. Uzasadnij jego działanie w kontekście przebiegów czasowych tego interfejsu

Urządzenia 1 wire potrafią być zasilane poprzez szynę danych. Urządzenie typu slave gromadzi moc w kondensatorze 900uF i dodatkowo jest podłączony z diodą zapobiegającą jego rozładowaniu

56. W której z podstawowych architektur systemów komputerowych można łatwo doprowadzić do wykonania programu w obszarze pamięci zarezerwowanej dla danych? wyjaśnij dlaczego.

Np. w Architekturze von Neumanna dlatego, że dane i kod programu przechowywany jest w tym samym obszarze pamięci. Pamięć komputerowa przechowuje dane programu oraz instrukcje programu; każda komórka pamięci ma unikatowy identyfikator nazywany jej adresem

57. Opisz w punktach, jaka jest sekwencja sygnałów podawanych na wejścia pamięci DRAM w cyklu odczytu komórki pamięci. Posługuj się konkretnymi jednoznacznymi nazwami magistral i pinów.

- 1) Na magistrali adresowej zostaje umieszczony numer wiersza zawierającego komórkę pamięci
- 2) Następnie sygnał RAS przechodzi w stan niski. Opadające zbocze RAS powoduje zapis numeru wiersza z magistrali adresowej w zatraskach połączonych z dekodern wierszy. Dekoder adresuje odpowiedni wiersz komórek, których zawartość zostaje przesłana do wzmacniaczy odświeżających, Komórki zostają odczytane i odświeżone - jeśli operacja dotyczy tylko odświeżania, to w tym momencie może się zakończyć
- 3) Na magistralę adresową trafia numer kolumny zawierającej komórkę. sygnał CAS przechodzi w stan niski- powoduje to zatrzaśnięcie numeru kolumny i przekazanie go do dekodera kolumn. Dekoder adresuje odpowiednią kolumnę wzmacniaczy odświeżających, które przekazują zawartość odczytanej komórki na magistralę danych.
- 4) Sygnały sterujące wracają do stanu neutralnego.

58. Opisz przeznaczenie oraz działanie mechanizmu “watchdog”, uwzględniając zarówno sposób jego realizacji od strony sprzętowej, jak i sposób użycia w kontekście programowym.

Watchdog: służy do naprawiania błędnie działającego systemu, bez pomocy człowieka i zapobiega awarii. W domyśle jest to układ zabezpieczający mikroprocesor przed zbyt długim przebywaniem w stanie zawieszenia. Mechanizm ten jest układem czasowym, który oczekuje na raporty o poprawnej pracy przez kontrolowane urządzenia co zadany czas. Brak potwierdzenia oznacza błąd i jest naprawiany przez restart, czasami przerwanie niemaskowalne lub chwilowe wyłączenie zasilania.

59. Czy adres fizyczny 5FC70h jest adresem początku segmentu?

Tak. Adres początku segmentu musi być podzielny przez 16 (0 na najmłodszej pozycji przy zapisie heksadecymalnym)

60. Czy adres fizyczny 4AA95h powstał z adresu logicznego 4555:5535h?

Nie. $4555h \cdot 16 + 5535h = 4AA85h$ W domyśle chodzi o tryb rzeczywisty, ale w wirtualnym jakby odpowiednio zamieszać może dałoby się taki adres uzyskać. Sposób obliczania adresów fizycznych jest w.

61. Czy adres fizyczny 5FC78h jest adresem początku segmentu? Nie.

62. W Z80CTC każdy kanał posiada swój własny rejestr wektora przerwań. Nie

63. Z80CTC nie posiada wejść bramkujących pozwalających na zablokowanie zliczania (w trybie licznikowym) w każdym kanale z osobna. Tak (takie wejścia ma za to Intel 8253).

64. Czy możliwe jest, aby Z80PIO zgłosił przerwania do procesora na skutek ustawienia określonego stanu na pojedynczym bicie portu A (pracującym jak wejście)? Tak, w trybie 3

65. W trybie wyjściowym opadające zbocze sygnału STB może spowodować wygenerowanie przerwania przez Z80PIO.

Nie. Przerwanie powoduje narastające zbocze STB

66. Czy w układzie Z80PIO każdy port, który może prowadzić transmisję z potwierdzeniem posiada swoją własną linię INT? Nie, linia INT jest tylko jedna.

67. W Z80PIO oba porty mogą pracować w trybie dwukierunkowym z potwierdzeniem. Nie, w trybie 2 może pracować tylko port A

68. W przypadku trzech układów Z80PIO podłączonych do procesora priorytet przerwań pomiędzy nimi jest ustawiany na podstawie wartości wpisywanej do rejestru sterującego przerwań Z80PIO. Nie, ustalany jest na podstawie połączeń wejść IEI i IEO.

69. W Z80PIO port B może pracować w trybie bitowym. Tak.

70. W trybie wyjściowym narastające zbocze sygnału STB może spowodować wygenerowanie przerwania przez Z80PIO. Tak.

71. W Z80PIO w trybie bitowym można wybrać, które linie portu A mają być liniami „we”, a które „wy”? Tak.

72. Adresy procedur obsługi przerwań do poszczególnych kanałów muszą być umieszczone kolejno w pamięci systemu.

Tak, ponieważ na wszystkie kanały jest jeden 8-bitowy rejestr z wektorem przerwania, w którym możemy zaprogramować bity V7-V3. D2-D1 to numer licznika (przy programowaniu są ignorowane), D0 = 0 (zawsze). Procedury przerwań muszą więc być w pamięci kolejno co 2 bajty.

73. W systemie komputerowym opartym na mikroprocesorze Z80 chcąc zwiększyć obciążalność magistrali procesora należy zastosować rejestry, które będą zachowywały zawartość magistrali. Tak

74. Z Z80PIO w trybie wejściowym sygnał STB jest generowany przez urządzenie zewnętrzne i służy do wpisania danej do rejestru Z80PIO. Tak

75. Aby Z80CTC pracował jako kontroler przerwań, w słowie sterującym musi być ustawiony odpowiedni bit uruchamiający ten tryb pracy. Nie, taki bit w ogóle nie istnieje.

76. W 8255 ustawianie i zerowanie bitów portu PB przy pracy w trybie 0 zachodzi poprzez, wpisanie odpowiedniego słowa sterującego do rejestru sterującego. Nie, po prostu wpisuje się do rejestru związanego z PB.

77. Czy w 8255 rejestr/bit sterujący określający możliwość zgłaszania przerwania przy pracy portu A w trybie wyjściowym z potwierdzeniem jest dostępny jako określony bit portu PC? Nie - nie są dostępne jako bity portu PC, chociaż można je kontrolować przez zapis do rejestru sterującego

78. W układzie 8255 port PC nie może pracować w trybie wyjściowym z potwierdzeniem. Tak, ponieważ brakuje linii potwierdzenia dla portu PC. Linię tę posiadają tylko porty A i B.

79. W układzie 8255 port PC może pracować tylko w trybie bez potwierdzenia. Tak, bo w trybie z potwierdzeniem jego bity są podkradane przez PA i PB.

80. Czy w układzie 8255 linia INT jest w razie potrzeby dostępna na jednej z linii portu PC? Tak.

81. W 8255 każdy z portów, które mogą pracować z potwierdzeniem ma w takim trybie pracy swoją własną linię INT.

Tak. (PC0 = INT B, PC3 = INT A)

82. Jednoczesne uaktywnienie się sygnałów \IORQ i \M1 na wejściach Z-80 PIO informuje, że procesor wystawił na magistralę adresową wektor obsługi przerwania. Nie. To urządzenie wystawia dolną część adresu miejsca w którym jest adres procedury obsługi przerwania na magistralę danych (nie adresową).

83. W przypadku łączenia układów peryferyjnych rodziny Z-80 (PIO, CTC, SIO i in.) w łańcuch daisy chain ich wyjścia INT nie mogą być ze sobą zwarte i podłączone razem do wejścia INT procesora Z-80. Nie. Wyjścia INT w takim przypadku zazwyczaj się zwiera i podłącza (z pull-upem) do wejścia INT procesora.

84. W 8255 ustawianie i zerowanie bitów wyjściowych portu PC przy pracy w trybie 0 zachodzi poprzez wpisanie odpowiedniego słowa sterującego do rejestru sterującego. Nie. Natomiast jest to prawda w trybach 1 i 2.

85. Napisz, w jaki sposób blokuje/odblokuje się zgłaszanie przerwania w porcie A lub B układu 8255 zaprogramowanego do pracy w trybie wyjściowym z potwierdzeniem. Wpisuje się do rejestru sterującego słowo w formacie:

D7 -- 0, D6--D4 - zarezerwowane, D3--D1 - numer bitu, D0 -- 1 (odblokowanie przerwania) lub 0 (zablokowanie). Numer bitu odpowiada przerzutników i INTE A lub INTE B

86. Czy w trakcie transmisji asynchronicznej może wystąpić przerwa trwająca np. 10 bitów? Tak (np. przerwa między kolejnymi znakami, bo między bitami oczywiście nie).

87. Czy procesor Z80 jest włączany wraz z innymi układami rodziny Z80 w łańcuch priorytetu przerwań (daisy chain) stanowiąc jednostkę nadrzędną w tym łańcuchu? Nie, nie jest on częścią łańcucha. Można powiedzieć, że jednostką nadrzędną jest pierwsze urządzenie w łańcuchu, mające wejście IEI podpięte (przez pull-up!) do VCC.

88. Priorytet przerwań w łańcuchu daisy chain jest ustalany poprzez fizyczne (w sensie połączeń elektrycznych) położenie każdego z układów (np. z rodziny Z80) względem innych? Tak (IEI, IEO).

89. Synchronizację transmisji asynchronicznej w łączu RS232 uzyskuje się dzięki linii przesyłającej sygnał zegarowy od komputera nadającego do odbierającego. Nie.

90. Czy procesor Z80 może rozróżnić w czasie przyjmowania przerwania od układu Z80 CTC (w drugim trybie przerwań - dedykowanym dla rodziny Z80), który kanał układu Z80 CTC spowodował zgłoszenie przerwania? Tak, jest to określone na drugim i trzecim bicie zwracanego wektora

91. Jeżeli w 8255 port A i B pracują w trybie wyjściowym z potwierdzeniem to port C można wykorzystać jako prosty 8-bitowy port wejściowy lub wyjściowy Nie. Linie portu C (nie wszystkie) są wykorzystywane do załatwienia potwierdzenia na portach A i B. Pozostałe, wolne bity można jednak wykorzystać.

92. Omówić budowę i zastosowania układów typu 8253.

Układ 8253 jest programowalnym licznikiem i często oznaczany jest, jako PIT (Programmable Interval Timer). System ten spełnia w komputerze zadanie zegara, generatora tonu, zapewniając przy tym prawidłowe (jeśli chodzi o czas) odświeżanie dynamicznego RAM-u. PIT (programmable interrupt - timer) zawiera trzy, niezależne od siebie 16-bitowe liczniki (liczące wstecz), wyposażone we własne wejścia - Gate oraz Clock, jak i wyjścia (OUT). Licznik 0 stosowany jest do generowania przerwania (IRQ 0), wykorzystywanego dla zegara systemowego w komputerach typu 8088/86. Odświeżanie dynamicznego RAM-u sterowane jest przez licznik 1, który (co 15 ms) przetwarza kanał DMA na 0. Licznik 2 odpowiedzialny jest za generowanie tonu i - w przeciwieństwie do innych liczników - może być również używany do innych celów. Stany liczników zapisywane są w trzech adresach rejestru (Counter-Address). Dla ich zaadresowania stosowane są kanały adresujące A0 i A1. Zapis i odczyt odbywa się za pomocą linii /WR oraz /RD. Układ uaktywniany jest przez ustawienie wartości Low na sygnale pinu /CS. Sygnał ten generowany jest przez odpowiedni układ dekodujący adresy. Przewody danych D0 do D7 połączone są bezpośrednio z magistralą danych, ponieważ bufor danych (Data Bus Buffer) może być ustawiony w stan wysokiej impedancji przez: /CS = High (Tri-State). Przez odpowiednie wejście GATE licznik jest uruchamiany, zatrzymywany lub cofany. Takt doprowadzany jest przez wejście CLK. W momencie, kiedy licznik osiąga wartość 0, na wyjściu pojawia się OUT.