

Josef Hloušek

**OSMIBITOVÉ
MIKROŘADIČE
ZILOG Z8**



Kniha se zabývá osmibitovými mikrořadiči řady Z8 firmy ZiLOG. Popisuje architekturu řady Z8, instrukční soubor, základy assembleru, ladění programů na hardwarovém emulátoru a práci s integrovaným vývojovým prostředím ZDS (ZiLOG Develop Studio). Obsahuje také popis jednoduché aplikace s obvodem Z86E02 a podrobně komentovaný výpis programu. Vlastnosti mikrořadičů, společně celé řadě Z8, jsou představeny prostřednictvím nejrozšířenějších obvodů Z86E02/04/08 a Z86E30/31/33/733. Za pomoci textu knihy je možno navrhnout jednoduchou aplikaci s obvody řady Z8 a postup pak uplatnit, po akceptování konkrétních parametrů, na kterýkoliv obvod řady Z8. Text knihy předpokládá, že čtenář je seznámen s obecnou teorií mikroprocesorů a mikrořadičů a ovládá používanou terminologii.

Josef Hloušek

Mikrořadiče ZiLOG Z8

Bez předchozího písemného svolení nakladatelství nesmí být kterákoli část kopírována nebo rozmnožována jakoukoli formou (tisk, fotokopie, mikrofilm nebo jiný postup), zadána do informačního systému nebo přenášena v jiné formě či jinými prostředky.

Autor a nakladatelství nepřejímají záruku za správnost tištěných materiálů. Předkládané informace jsou zveřejněny bez ohledu na případné patenty třetích osob. Nároky na odškodnění na základě změn, chyb nebo vynechání jsou zásadně vyloučeny.

Všechny registrované nebo jiné obchodní známky použité v této knize jsou majetkem jejich vlastníků. Uvedením nejsou zpochybněna z toho vyplývající vlastnická práva.

Veškerá práva vyhrazena

© Ing. Josef Hloušek, Praha 1999

Nakladatelství BEN – technická literatura, Věšínova 5, Praha 10

Josef Hloušek: Mikrořadiče ZiLOG Z8

BEN – technická literatura, Praha 1999

1. vydání

ISBN 80-86056-81-3

OBSAH

1.	VŠEOBECNÉ PARAMETRY MIKROŘADIČŮ Z8	8
1.1	Mikrořadiče typu Z86 E30, Z86 E31, Z86 E33, Z86 733	8
1.2	Mikrořadiče typu Z86 E02, Z86 E04, Z86 E08	9
1.3	Elektrické parametry mikrořadičů Z8	10
2.	ADRESNÍ PROSTOR MIKROŘADIČŮ Z8	11
2.1	Paměť RAM	11
2.1.1	Standardní sada registrů	12
2.1.2	Rozšířená sada registrů, banka F	13
2.2	Paměť PROM	13
2.3	Zásobník (Stack)	14
2.4	Řídící registry pro přístup do paměti	15
2.5	Registry, jejich adresy a identifikátory	16
3.	OSCILÁTOR MIKROŘADIČŮ Z8	18
4.	RESET mikrořadičů Z8	20
4.1	Definovaný stav řídicích registrů a portů	20
4.2	RESET při zapnutí napájení (POR – Power On Reset)	21
4.3	WDTMR – Watch Dog Timer	21
5.	VSTUPNÍ/VÝSTUPNÍ PORTY	23
5.1	Port 0	24
5.2	Port 2	26
5.3	Port 3	27
5.4	Analogové komparátory	28
5.5	Režim sníženého vyzařování rušivých napětí	29
6.	ČÍTAČE, ČASOVAČE	30
6.1	Řízení čítačů	32
6.2	Výstupní módy čítačů (pouze u typů Z86E30/31/33/733)	33
6.3	Vstupní módy čítače T1	34
6.3.1	Počítání vnějších impulzů	34
6.3.2	Hradlování vnitřního signálu SCLK vnějšími impulzy	34
6.3.3	Monostabilní režim jednorázový	35
6.3.4	Monostabilní režim opakovatelný	35
6.3.6	Nastavení čítačů po signálu RESET	36
6.3.5	Zapojení čítačů T0 a T1 do kaskády	36
7.	PŘERUŠENÍ	37
7.1	Zdroje přerušení	37
7.2	Priorita přerušení	38
7.3	Maskování přerušení	39
7.4	Programové přerušení	39

7.5	Vektorové přerušení	40
7.6	Přerušení na dotaz	40
8.	ÚSPORNÉ REŽIMY	42
8.1	Režim HALT	42
8.2	Režim STOP	42
8.2.1	Ukončení režimu STOP signálem Power On Reset	43
8.2.2	Ukončení režimu STOP časovačem WDT	43
8.2.3	Ukončení režimu STOP signálem z určeného zdroje	43
9.	REGISTR VOLITELNÝCH FUNKCÍ	45
10.	PROGRAMOVÁNÍ MIKROŘADIČŮ Z8	47
10.1	Příznak výsledku operace	47
10.2	Notace používaná pro popis instrukcí	49
10.3	Čtyřbitová adresa na místě osmibitové adresy	49
10.4	Instrukční soubor Z8	50
11.	ZMASM – ZILOG MACRO CROSS ASSEMBLER	70
11.1	Historie ZMASM	70
11.2	Struktura ZMASM	71
11.3	Formát zdrojového textu pro ZMASM	73
11.3.1	Příkazový řádek	73
11.3.2	Pole návěští (label)	74
11.3.3	Pole instrukce	74
11.3.4	Pole operandu	74
11.3.5	Pole komentáře	74
11.3.6	Konstanty	75
11.3.7	Symboly	76
11.3.8	Operátory	76
11.3.9	Výrazy	77
11.4	Direktivy assembleru	77
12.	VÝVOJOVÉ PROSTŘEDÍ ZILOG	81
12.1	Sestava vývojového prostředí pro mikrořadiče Z8	81
12.2	Napájecí zdroj pro emulátor	82
12.3	Propojení emulátoru s osobním počítačem	82
12.4	Přizpůsobení emulátoru pro požadovaný typ mikrořadiče	82
12.5	Přizpůsobení emulátoru k umístění krystalu oscilátoru	83
12.6	Přizpůsobení emulátoru ke způsobu napájení aplikace	83
12.7	Připojení emulátoru k aplikaci	84
12.8	Uživatelsky důležité prvky na desce emulátoru	84
12.9	Úprava emulátoru Z86CCP00ZEM	85
12.10	Spuštění emulátoru	85
12.10.1	Instalace GUI	85
12.10.2	Spuštění GUI	85

13.	ZiLOG DEVELOPER STUDIO ZDS 2.11	87
13.1	Historie ZDS	87
13.2	Instalace ZDS 2.11	87
13.3	Spuštění ZDS	88
13.4	Přehled funkcí ZDS	89
13.4.1	Menu File	90
13.4.2	Menu Edit	90
13.4.3	Menu View	91
13.4.4	Menu Project	91
13.4.5	Menu Build	92
13.4.6	Menu Tools	92
13.4.7	Menu Window	92
13.4.8	Menu Help	93
13.5	Konfigurace projektu a překlad zdrojového textu	93
13.5.1	Založení projektu	94
13.5.2	Sestavení projektu	95
13.5.3	Konfigurace assembleru pro projekt	95
13.5.4	Konfigurace linkeru pro projekt	96
13.5.5	Vytvoření spustitelného souboru *.ld	97
13.6.1	Konfigurace emulátoru	99
13.6.2	Spuštění komunikace s emulátorem	99
13.6.3	Spuštění programu	100
13.6.4	Nastavení zářezek (break point)	101
13.6.5	Ladění programu	102
13.6.6	Zobrazení obsahu registrů při ladění	102
13.6.7	Opravy zdrojového textu při ladění	103
13.6.8	Uzavření a znovuotevření rozpracovaného projektu	104
13.7	Programování OTP verzí mikrořadičů	104
13.7.1	Konfigurace programování	104
13.7.2	Vložení programu do OTP paměti mikrořadiče	106
14.	MĚŘIČ POHOTOVÉ REAKCE	
	S MIKROŘADIČEM Z86E02 08 PSC	107
14.1	Obvodové řešení měřiče	107
14.2	Funkce měřiče	108
14.3	Výpočet konstant pro programování čítače T1	110
14.3.1	Měření reakční doby	110
14.3.2	Astabilní multivibrátor s periodou cca 0,5 s (výchozí stav)	111
14.4	Popis programu	111
14.5	Mapa symbolů v paměti RAM	112
14.6	Výpis programu reaction.s	113
	LITERATURA a INTERNETOVÉ ADRESY	122
	Knihy nakladatelství BEN – technická literatura	123, 126
	Jaké jsou důvody pro aplikace mikrořadičů ZiLOG Z8	124
	Sortiment a adresa firmy ECOM s. r. o.	125
	Kontaktní adresy firmy BEN – technická literatura	127

HISTORIE OSMIBITOVÝCH MIKROPROCESSORŮ A MIKROŘADIČŮ ZILOG

Americká firma ZiLOG vstoupila na trh mikroprocesorů v roce 1973. V době, kdy svět dobývaly obvody Intel 8080, se objevil obvod s typovým označením Z80. Osmibitový mikroprocesor s odlišnou architekturou, založenou na sadě univerzálních registrů, které mohly sloužit pro jakoukoliv funkci, tedy i jako střadač. Architektura umožňovala pružnější přístup k registrům, optimální využití instrukcí a tím i rychlejší chod programu než u procesorů jiných výrobců při stejném hodinovém kmitočtu.

Ještě dříve, než IBM dala světu osobní počítače, museli se spokojit první počítačová nadšenci s čímsi, co je dnes možno srovnávat s PC jen velmi stěží. Dodnes však jméno Sinclair představuje pojem. A Sinclair obsahoval mikroprocesor Z80 CPU !

Postupem času byl mikroprocesor Z80 CPU (Central Processor Unit) následován obvody Z80 CTC (Counter-Timer Circuit), Z80 PIO (Parallel Input/Output), Z80 SIO (Serial Input/Output), Z80 DMA (Direct Memory Access) a dalšími. S rozvojem technologie se jednak měnila struktura obvodů od TTL přes NMOS až po CMOS, jednak se zvětšovala hustota integrace a v jednom pouzdře se sloučovalo stále více obvodů dřívější stavebnice. Na konci vývoje stál obvod Z80 IPC (Intelligent Peripheral Controller), který obsahoval nejen procesor, ale paralelní i sériové vstupně/výstupní obvody (Porty), oscilátor, čítače. Tedy podle dnes zavedené terminologie – mikrořadič. Obvody řady Z80, přestože dnes již typování obvodů smazalo příslušnost k řadě Z80 (např. Z84C0010PEC je novodobý CMOS procesor Z80), tvoří značný podíl na světové produkci mikroprocesorů.

Až na výjimky, obvody řady Z80 nikdy neobsahovaly žádnou paměť kromě základní registrové výbavy. Ani současní pokračovatelé řady, obvody Z180 a šestnáctibitový Z380 nemají na čipu paměť. To je v tvrdé konkurenci nevýhoda. Musel se objevit nový obvod.

Obvody řady Z8 jsou 8bitové CMOS mikrořadiče. Vycházejí z principu univerzální sady registrů, na čipu je však těchto registrů pro všeobecné použití (General Purpose Register) až 256, takže už je možno hovořit o paměti RAM. Navíc čip obsahuje i paměť pro uložení programu v provedení PROM (používá se zkratka OTP – One Time Programmable) a to nejméně 512 Bytů ale také až 32 KBytů. Dále čip obsahuje dva 14bitové čítače, dva analogové komparátory, oscilátor a nejméně 14 vstupů/výstupů. Čip dále obsahuje Watch Dog Timer (překlad tohoto termínu do češtiny se stále nedaří), Power On Reset (český termín nulování při zapnutí napájení je možný, ale anglický je lepší) a další obvody.

Všechny obvody řady Z8 mají jeden shodný instrukční soubor. Mnemonicky připomíná instrukční soubor Z80, funkčně je však jiný. Přesto přechod od programování obvodů Z80 k řadě Z8 je snadný. Pro efektivní programování je k dispozici assembler, pro podporu vývoje je používán hardwarový emulátor řízený osobním počítačem.

Řada Z8 v roce 1999 obsahuje více jak stovku jednotlivých typů, lišících se velikostí paměti, počtem vstupů a výstupů, rychlostí, dovolenou pracovní teplotou, pouzdry pro klasickou i povrchovou montáž. Speciální obvody obsahují i AD a DA převodníky, sériový komunikační kanál (UART), jádro Z8 tvoří základ mnoha jednoúčelových obvodů pro řízení klávesnic osobních počítačů nebo obvodů pro infračervené dálkové ovládání.

Tato učebnice nemůže a ani nechce popsat funkci všech obvodů řady Z8. Chce pouze na několika málo obvodech vysvětlit vlastnosti společné celé řadě Z8. Učebnice popisuje architekturu řady Z8, instrukční soubor, práci s assemblerem a emulátorem. S pomocí učebnice je tedy možno navrhnout aplikaci s nejrozšířenějšími obvody řady Z8 a postup pak uplatnit, po akceptování konkrétních parametrů, na kterýkoliv obvod řady Z8.

Text učebnice předpokládá, že čtenář je seznámen s obecnou teorií mikroprocesorů a mikrořadičů a ovládá používanou terminologii. Autor se omlouvá za častější používání originálních anglických termínů všude tam, kde se jejich používání vžilo, kde český překlad je méně výstižný nebo směšný a především tam, kde je to nutné (procesor bude vždy rozumět pouze instrukci DJNZ – Decrement and Jump if Non Zero a nikoliv ZSNN – Zmenši o jednu a Skákej když Není Nula).