

Arhitectura hardware a kitului RCM3365.

Microprocesorul Rabbit 3000

1. Scopul lucrării

Scopul acestei lucrări de laborator este familiarizarea studenților cu arhitectura hardware a modului RCM3365 și a microprocesorului Rabbit 3000. Se vor studia subsistemele modului RCM3365, pinii acestui modul și funcțiile acestora, precum și arhitectura internă a microprocesorului Rabbit 3000.

2. Arhitectura hardware a modului RCM3365

În continuare vor fi descrise toate componentele și subsistemele hardware ale modulelor RCM3365. În figura următoare este prezentată schema modului RCM3365, cu toate subsistemele sale hardware.

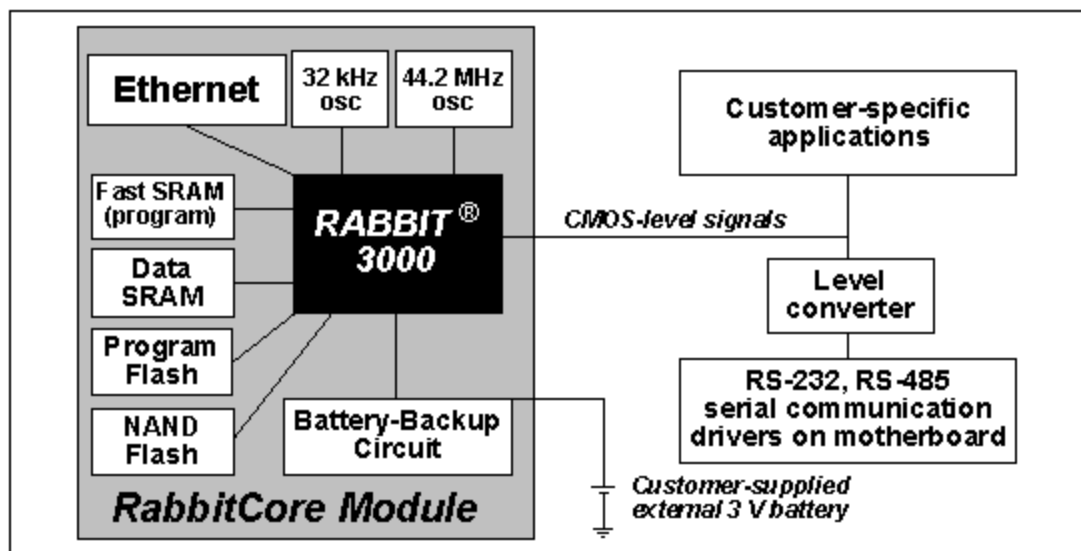


Figura 1. Subsistemele modului RCM3365

2.1. Intrările și ieșirile modului RCM3365

Modulul RCM3365 se conectează la placa de bază prin intermediul a doi conectori J3 și J4 standard 2 x 34 pini cu distanța între pini de 2 mm. În figura următoare sunt prezentați cei doi conectori J3 și J4, împreună cu semnificația pinilor.

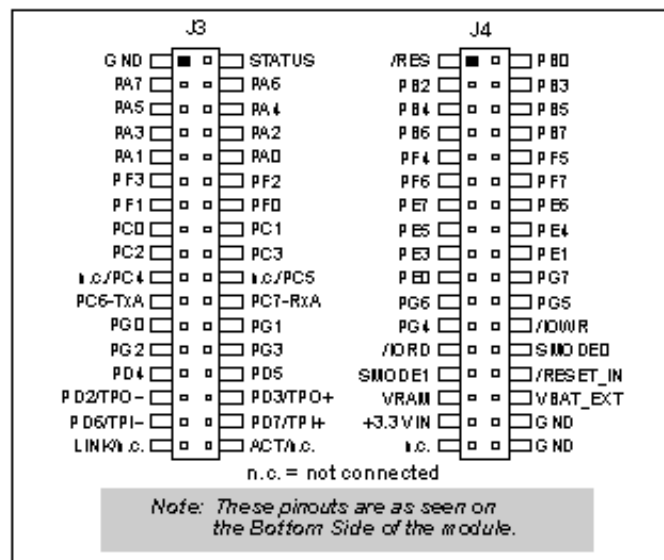


Figura 2. Pinii modulului RCM3365

Pinul PB0 este utilizat pentru a detecta prezența cardului de memorie xD-Picture. Pinii 29-32 de pe conectorul J3 sunt configurați folosind ștrapuri în pozițiile JP4, JP5, JP7 și JP8 ca fiind semnalele PD3, PD2, PD7 și, respectiv, PD6. Mai pot fi configurați să transporte semnalele Ethernet TPO+, TPO-, TPI+ și TPI-. Pinii 33 și 34 de pe conectorul J3 sunt legați la semnalele LINK și ACT, care vor aprinde LED-urile corespunzătoare de pe modulul RCM3365. Aceste semnale pot fi deconectate prin înlăturarea rezistorilor R41 și R42 de pe modul (vezi figura 3).

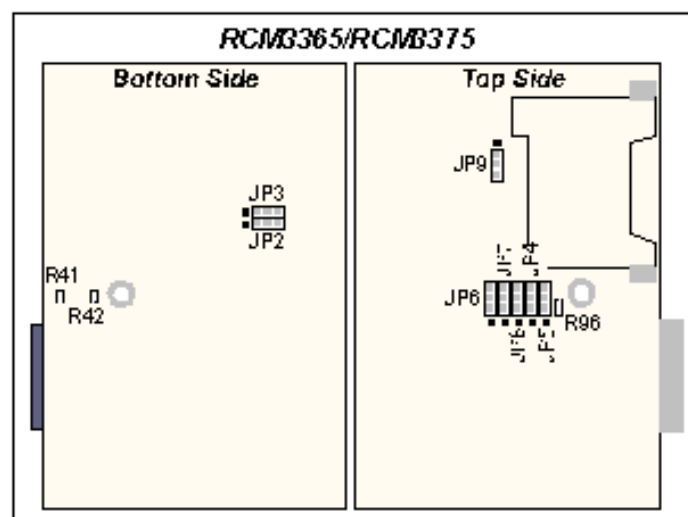


Figura 3. Pozițiile jumperilor de configurare ai modulului RCM3365

În figura 3 sunt prezentate pozițiile jumperilor de configurare ai modulului RCM3365. Micile pătrate înnegrite semnalizează poziția pinului 1 al jumperului.

În tabelul 1 sunt prezentate pozițiile jumperilor de configurare și semnificația acestora.

Conector	Descriere	Pini conectați		Poziție implicită
JP2	Selecție Bank de memorie Flash	1-2	Mod Bank	
		2-3	Mod normal	X
JP3	Dimensiune SRAM Date	1-2	128K/256K	
		2-3	512K	X
JP4	Ethernet sau leșire pe conectorul J3	1-2	TPO+	
		2-3	PD3	X
JP5	Ethernet sau leșire pe conectorul J3	1-2	TPO-	
		2-3	PD2	X
JP6	Ethernet sau leșire pe conectorul J3	1-2	ENET_INIT	
		2-3	PE0	X
JP7	Ethernet sau leșire pe conectorul J3	1-2	TPI+	
		2-3	PD7	X
JP8	Ethernet sau leșire pe conectorul J3	1-2	TPI-	
		2-3	PD6	X
JP9	Semnalele de selecție a circuitului pentru memoria Flash NAND și cardul xD-Picture	1-2	Semnale de selecție separate către memoria Flash NAND și cardul de memorie xD-Picture	
		2-3	Semnale de selecție separate pentru memoria Flash NAND și cardul de memorie xD-Picture	X
R96	Detecție card xD-Picture	instalat	Detecție card xD-Picture existentă	X
		neinstalat	PB0 poate fi folosit ca CLKB (Portul Serial sincron B)	

Tabelul 1. Configurația jumperilor modulului RCM3365

Porturile microprocesorului Rabbit 3000 folosit în cadrul modulului RCM3365 sunt configurabile, deci setările implicite din fabrică pot fi schimbate. În figura 4 sunt prezentate porturile microprocesorului Rabbit 3000 utilizate în modulul RCM3365, iar în tabelul 2 sunt prezentate setările implicite din fabrică și alternativele acestora.

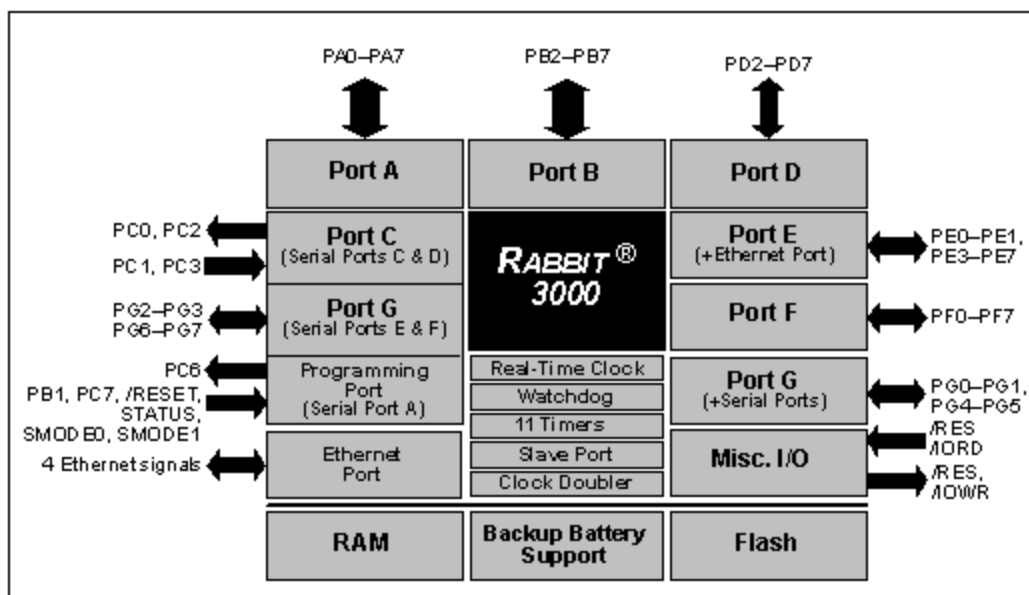


Figura 4. Porturile microprocesorului Rabbit 3000

Tabelul 2. Configurațiile pinilor modului RCM3365

PIN	NUME PIN	FUNCȚIE IMPLICITĂ	FUNCȚIE ALTERNATIVĂ	NOTE
Conectorul J3	1	GND		
	2	STATUS	leșire (Status)	leșire
	3-10	PA [7:0]	Intrare/leșire paralelă Magistrală de date externă (ID0 – ID7) Magistrală de date pentru portul slave (SD0 – SD7)	Magistrala de date externă
	11	PF3	Intrare/leșire	QD2A
	12	PF2	Intrare/leșire	QD2B
	13	PF1	Intrare/leșire	QD1A CLKC
	14	PF0	Intrare/leșire	QD1B CLKD
	15	PC0	leșire	TXD
	16	PC1	Intrare	RXD
	17	PC2	leșire	TXC
	18	PC3	Intrare	RXC
	19	PC4	leșire	TXB
	20	PC5	Intrare	RXB
	21	PC6	leșire	TXA
	22	PC7	Intrare	RXA
	23	PG0	Intrare/leșire	TCLKF
	24	PG1	Intrare/leșire	RCLKF
	25	PG2	Intrare/leșire	TXF
	26	PG3	Intrare/leșire	RXF
	27	PG4	Intrare/leșire	ATXB
	28	PD5	Intrare/leșire	ARXB
	29	PD2 / TPO–	Intrare/leșire	TPOUT– *
	30	PD3 / TPO+	Intrare/leșire	TPOUT+ *
	31	PD6 / TPI–	Intrare/leșire	TPIN– *
	32	PD7 / TPI+	Intrare/leșire	TPIN+ *
	33	LINK	leșire	Curentul maxim 1 mA (vezi obs. 1)
	34	ACT	leșire	

* Pini 29-32 sunt configurați cu jumperi la JP4, JP5, JP7 și JP8.

Conectorul J4

PIN	NUME PIN	FUNCȚIE IMPLICITĂ	FUNCȚIE ALTERNATIVĂ	NOTE	
Conectorul J4	1	/RES	leșire reset	leșirea din generatorul de reset	
	2	PB0	Intrare/leșire	CLKB	Folosit pentru a detecta prezența cardului xD – Picture
	3	PB2	Intrare/leșire	IA0 /SWR	Adresă externă 0 Scriere port slave
	4	PB3	Intrare/leșire	IA1 /SRD	Adresă externă 1 Citire port slave
	5	PB4	Intrare/leșire	IA2 SA0	Adresă externă 2 Adresă 0 port slave
	6	PB5	Intrare/leșire	IA3 SA1	Adresă externă 3 Adresă 1 port slave
	7	PB6	Intrare/leșire	IA4	Adresă externă 4
	8	PB7	Intrare/leșire	IA5 /SLAVEATTN	Adresa externă 5 Atenție Slave
	9	PF5	Intrare/leșire	AQD1B PWM0	
	10	PF5	Intrare/leșire	AQD1A PWM1	
	11	PF6	Intrare/leșire	AQD2B PWM2	
	12	PF7	Intrare/leșire	AQD2A PWM3	
	13	PE7	Intrare/leșire	I7 /SCS	
	14	PE6	Intrare/leșire	I6	
	15	PE5	Intrare/leșire	I5 INT1B	
	16	PE4	Intrare/leșire	I4 INT0B	
	17	PE3	Intrare/leșire	I3	
	18	PE1	Intrare/leșire	I1 INT1A	Strobe 1 I/O Întreruperea 1A
	19	PE0	Intrare/leșire	I0 INT0A	Strobe 0 I/O Întreruperea 0A
	20	PG7	Intrare/leșire	RXE	Portul serial E
	21	PG6	Intrare/leșire	TXE	
	22	PG5	Intrare/leșire	RCLKE	Intrarea ceasului serial E
	23	PG4	Intrare/leșire	TCLKE	leșirea ceasului serial E
	24	/IOWR	leșire		Strobe scriere externă
	25	/IORD	Intrare		Strobe citire externă
	26 - 27	SMODE0, SMODE1	(0,0) – începe execuția la adresa zero (0,1) – boot rece de la portul slave (1,0) – boot rece de la Portul Serial sincron A SMODE0=1, SMODE1=1 Boot rece de la Portul Serial asincron A cu viteza de 2400 bps (cablul de programare este conectat)		De asemenea conectat la cablul de programare
	28	/RESET_IN	Intrare		Intrarea către Generatorul de Reset
	29	VRAM	leșire		Vezi observațiile de sub tabel
	30	VBAT_EXT	Intrare baterie 3 V		Tensiunea minimă a bateriei 2.85 V
	31	+3.3 VIN	Intrare Tensiune		3,15 – 3,45 V _{CC}
	32	GND			

33	N.C.			Rezervat folosirii ulterioare
34	GND			

Observații:

1. La folosirea pinilor 33-34 de pe conectorul J3 pentru a alimenta LED-urile, acești pini pot suporta un curent maxim de până la 8mA.
2. Tensiunea VRAM este dependentă de temperatură. Dacă tensiunea VRAM scade sub 1,2V până la 1,5V, conținutul SRAM-ului alimentat de baterie ar putea fi pierdut. Dacă VRAM scade sub 1,0V, oscilatorul de 32 KHz s-ar putea opri. Acordați atenție acestei tensiuni dacă folosiți curent de pe acest pin.

2.1.1. Interfața memorie I/O

Liniile de adrese ale procesorului Rabbit 3000 (A0–A18) și toate liniile de date (D0–D7) sunt rutate intern către memoria flash și circuitele de memorie SRAM. Semnalele /IOWR și /IORD sunt disponibile pentru conectarea cu dispozitivele externe – fiți atenți la încărcarea acestor două semnale dacă le folosiți, fiindcă acestea sunt de asemenea folosite de către modulul RCM3365.

Portul paralel A poate fi folosit ca magistrală I/O externă de date pentru a izola I/O extern de magistrala de date principală. Pinii portului paralel B PB2–PB7 pot fi folosiți ca o magistrală de adrese auxiliară.

Când folosiți magistrala I/O auxiliară pentru ieșire digitală sau modulul tastatură/LCD pe placa de bază, sau pentru orice alt motiv, trebuie să adăugați următoarea linie la începutul programului:

```
#define PORTA_AUX_IO // necesară pentru a valida magistrala I/O auxiliară
```

Ieșirea STATUS are trei funcții diferite programabile:

1. Poate fi setată jos la primul ciclu de fetch al opcodului
2. Poate fi setată jos în timpul unui ciclu de recunoaștere a întreruperii
3. Poate fi folosită ca ieșire de uz general.

2.1.2. Alte intrări și ieșiri

Doi pini de stare, SMODE0 și SMODE1, sunt disponibili ca intrări. Starea logică a celor doi pini determină procedura de pornire după resetare.

/RESET_IN este o intrare externă folosită pentru a reseta microprocesorul Rabbit 3000 și circuitele periferice de pe modulul RCM3365. /RES este o ieșire din circuitul de reset care poate fi folosită pentru resetarea dispozitivelor periferice externe.

2.1.3. LED-uri

RCM3365 are cinci LED-uri localizate în jurul conectorului RJ-45 Ethernet: **ACT**, **LINK**, **SPEED**, **FM**, și **USR**.

LED-ul galben **ACT** de la DS1 indică activitatea rețelei.

LED-ul verde **LINK** de la DS2 indică faptul că modulul RCM3365 este conectat la o rețea funcțională.

LED-ul verde **SPEED** de la DS4 este folosit pentru a indica atunci când modulul RCM3365 este conectat la o conexiune Ethernet 100Base-T.

Ledul **FM** de la DS3 clipește atunci când informația este scrisă sau citită de la dispozitivul de stocare Flash.

Ledul roșu **USR** de la DS3 este un LED programabil de către utilizator care este controlat de pinul PD0 de pe portul paralel D al microprocesorului Rabbit 3000. Programul demonstrativ FLASHLED.C ce se găsește în directorul Dynamic C \SAMPLES\RCM3360 prezintă modul de setare și de utilizare a acestui LED programabil.

2.2. Comunicații seriale

Modulul RCM3365 nu are transmițătoare seriale direct pe placă. Totuși, o interfață serială poate fi încorporată în placa de bază. De exemplu, pe placa de bază Rabbit sunt montate transmițătoarele RS-232 și RS-485.

2.2.1. Porturi seriale

Sunt 6 porturi seriale denumite porturile seriale A, B, C, D, E și F. Toate 6 porturi pot opera asincron cu viteze de până la ceasul sistemului divizat cu 8. Un port asincron poate opera cu 7 sau 8 biți de date. Mai este suportată și o schemă cu 9 biți, unde bitul adițional marchează primul octet dintr-un mesaj.

Portul serial A este folosit în mod normal ca port de programare, dar poate fi folosit ca un port asincron sau ca port serial sincron odată ce modulul RCM3365 a fost programat și operează în modul Run.

Portul serial B este disponibil pe RCM3365 și poate fi folosit ca un port asincron. PB0 este folosit pentru a detecta dacă este instalat un card de memorie xD-Picture. Dacă nu este folosită detecția cardului de către aplicație, se poate scoate R96 (vezi figura 3) pentru a dezactiva detecția cardului de memorie xD-Picture și atunci se poate folosi portul B ca un port serial sincron.

Porturile seriale C și D pot fi folosite de asemenea în modul sincron. În acest mod un ceas sincronizează informațiile de intrare sau ieșire. Ceasul poate fi furnizat de către oricare din aparatele care comunică.

Și porturile seriale E și F pot fi configurate ca porturi seriale HDLC. Protocolul IrDA este suportat în formatul SDLC de către aceste două porturi.

2.2.2. Portul Ethernet

Figura 5 prezintă pinii portului Ethernet RJ-45. De notat că unii conectori Ethernet sunt numerotați invers ca aici.

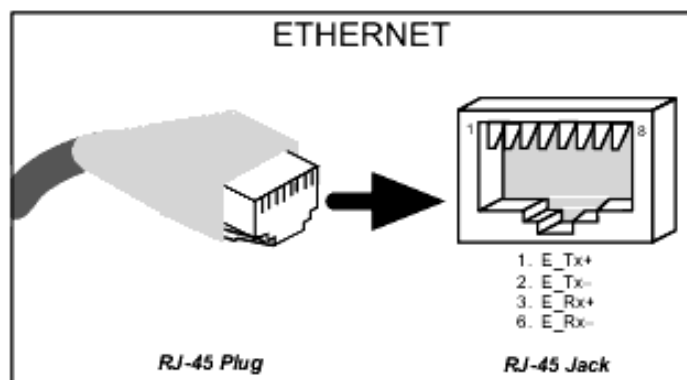


Figura 5. Pinii portului Ethernet RJ-45

Trei LED-uri sunt amplasate lângă conectorul Ethernet RJ-45, unul pentru a identifica legătura Ethernet (LINK), unul pentru a indica activitatea Ethernet (ACT) și unul pentru a indica viteza 10/100 Base T (SPEED).

Conectorul RJ-45 este ecranat pentru a minimiza efectele interferenței electromagnetice de la/către semnalele Ethernet.

2.2.3. Portul de programare

Portul serial A are trăsături speciale care permit acestuia repornirea după ce sistemul a fost resetat. Portul A este de asemenea portul folosit pentru dezvoltarea de aplicații sub Dynamic C. Modul RCM3365 este accesat folosind un conector de programare intitulat J1 sau prin intermediul conectorului Ethernet. Portul de programare folosește portul serial A al microprocesorului Rabbit 3000 pentru comunicații și este folosit pentru următoarele operații:

- Programare / depanare
- Clonare
- Programare / depanare de la distanță prin conexiunea Ethernet

Când este folosit conectorul J1, pinii de stare la pornire (SMODE0, SMODE1) apar pe portul de programare, astfel încât un dispozitiv extern să poată forța placa să pornească în modul bootstrap extern. Placa poate fi resetată de către Dynamic C prin intermediul liniei /RESET din conectorul J1.

Pinul de stare al procesorului Rabbit 3000 este de asemenea prezent în portul de programare. Pinul de stare este o ieșire ce poate fi folosită pentru a trimite un semnal digital general.

Linia ceasului pentru portul serial A apare și ea în portul de programare, ceea ce face posibilă comunicarea serială sincronă.

Portul de programare este folosit pentru a porni modulul RCM3365 într-un mod în care acesta va descărca un program de la port și apoi îl va executa. Portul de programare transmite informații de la / către calculator în timp ce un program este depanat.

Portul de programare poate fi folosit ca un port aplicație prin intermediul conectorului J1 dacă se folosește conectorul DIAG de pe cablul de programare. Portul de programare poate fi folosit ca:

- un port serial sincron
- un port serial asincron, cu linia de ceas folosită ca o intrare CMOS de uz general
- două intrări CMOS de uz general și o ieșire CMOS de uz general.

2.3. Cablul de programare

RCM3365 este trecut automat în modul “programare” când conectorul PROG de pe cablul de programare este conectat și este trecut automat în modul “run” când nici un cablu de programare nu este conectat.

Conectorul DIAG al cablului de programare poate fi utilizat pe conectorul J1 al modulului RCM3365, acesta lucrând în modul “run”. Acest lucru face posibil ca portul de programare să fie utilizat ca un port serial oarecare.

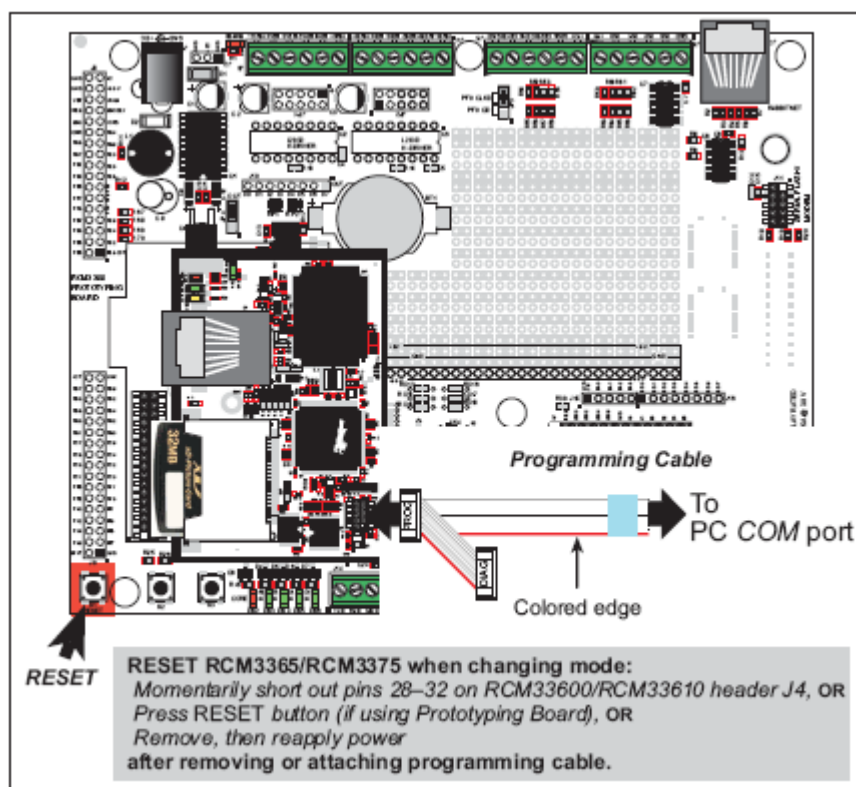


Figura 6. Trecerea din modul Programare în modul Run

2.3.1. Trecerea din modul Programare în modul Run

1. Deconectați cablul de programare din conectorul J1 al modulului RCM3365.
2. Resetați RCM3365. Puteți face acest lucru urmând instrucțiunile din figura 6. RCM3365 este acum gata să lucreze în modul Run.

2.3.2. Trecerea din modul Run în modul Programare

1. Conectați cablul de programare în conectorul J1 al modulului RCM3365.
2. Resetați RCM3365. Puteți face acest lucru urmând instrucțiunile din figura 6. RCM3365 este acum gata să lucreze în modul Programare.

2.4. Memoria

2.4.1. SRAM

Modulele RCM3365 au la U11 512K de memorie rapidă SRAM destinată execuției programelor. Această memorie nu este alimentată (asigurată) pe baterie. Există, însă, instalată la U10 o memorie de date de 512K care este asigurată pe baterie.

2.4.2. Flash EPROM

Modulele RCM3365 au 512K de memorie EPROM Flash la locația U9.

Observație: Z-World recomandă ca nici o aplicație a clientului să nu fie constrânsă de dimensiunea sectorului de memorie al Flash EPROM-ului, din moment ce ar putea să apară o schimbare a dimensiunii sectorului în viitor.

Scrierea către adrese aleatoare ale memoriei flash în timpul execuției programului este, de asemenea, nerecomandată. Ca alternativă, folosiți o parte din zona utilizator pentru a stoca date persistente. Pentru acest scop sunt furnizate funcțiile `writeUserBlock` și `readUserBlock`.

2.4.3. NAND Flash

RCM3365 suportă carduri de memorie portabilă *xD-Picture* (de până la 128MB) pentru a stoca date și pagini Web și mai este dotat și cu 16MB de memorie NAND Flash pe placă.

Observație: Sistemele bazate pe Rabbit nu au implementat specificația *xD-Picture* pentru stocare de date și nici nu sunt compatibile cu dispozitivele de citire a cardurilor *xD-Picture*.

Memoria NAND Flash și cardul *xD-Picture* sunt potrivite pentru aplicații de stocare în masă, dar neadekvate, în general, execuției directe a programului. Memoriile NAND Flash diferă de memoriile NOR Flash paralel (tipul de memorie flash folosit pentru stocarea de cod pe circuitele Rabbit și modulele RabbitCore aflate în producție în prezent) prin două aspecte. Mai întâi, memoria NAND Flash are nevoie de cod corector de erori (ECC) pentru fiabilitate. Deși producătorii de NAND Flash garantează că blocul 0 va fi fără erori, cei mai mulți

producători garantează că un circuit de memorie NAND Flash nou va fi livrat cu un procent relativ mic de erori și că acesta nu va depăși un număr maxim sau un procent de erori pe timpul duratei sale de viață de până la 100000 de scrieri. În al doilea rând, metodele standard de adresare a memoriei NAND Flash multiplexează comenzi, date și adrese pe aceiași pini I/O, timp în care cer ca anumite linii de control să fie păstrate stabile pe durata accesului la memoria NAND Flash. Apelurile de funcții software furnizate de către Z-World pentru NAND Flash se asigură de integritatea datelor și a atributelor de fiabilitate.

Figura 7 prezintă modul de introducere/extragere a cardului de memorie *xD-Picture*. Pe durata introducerii/extragerii cardului, aveți grijă să nu atingeți contactele electrice de pe capătul acestuia, pentru a preveni alterarea prin descărcare electrostatică a cardului și pentru a ține departe de contacte umezeala sau alți factori dăunători. Nu introduceți/extrageți cardul în timp ce acesta este accesat de către circuit.

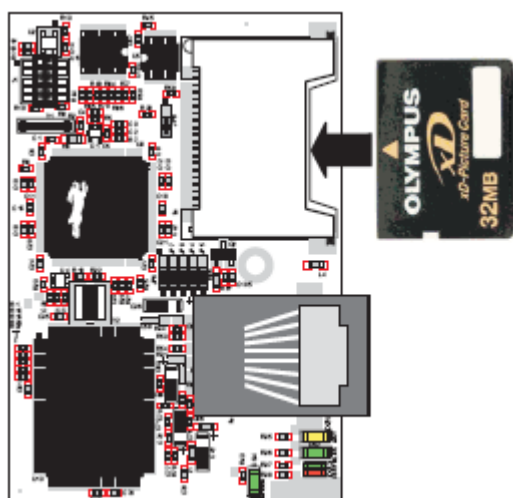


Figura 7. Introducerea/extragerea cardului de memorie xD-Picture

Este posibilă extragerea cardului de memorie *xD-Picture* fără a opri alimentarea circuitului RCM3365. Sistemul de fișiere trebuie închis înainte ca această operațiune să fie realizată. Trebuie trecuți în starea inactivă toți pinii asociați cu memoria NAND Flash și cu cardul *xD-Picture* și trebuie întrerupte orice operațiuni de citire/scriere din/în zona NAND Flash. Aceste operațiuni pot fi inițiate în software prin detectarea activării unui switch extern de către utilizator, iar atunci cardul xD-Picture poate fi extras și înlocuit cu un altul. Odată ce aplicația detectează un nou card, sistemul de fișiere poate fi deschis. Acești pași permit instalarea sau deinstalarea cardurilor xD-Picture fără a afecta nici programul, care continuă să ruleze pe modulul RCM3365, nici datele stocate pe cardul de memorie xD-Picture.

2.5. Alte componente hardware

2.5.1. Dublorul de ceas

RCM3365 se bucură de dublorul de ceas intern al microprocesorului Rabbit 3000. Acest dublor de ceas încorporat permite utilizarea cristalelor cu jumătate de frecvență pentru

a reduce emisiile radiate. Frecvența de 44,2 MHz specificată pentru RCM3365 este generată cu ajutorul unui rezonator de 22,12 MHz.

Dublorul de ceas poate fi dezactivat dacă viteze de ceas de 44,2 MHz nu sunt necesare. Aceasta va reduce consumul de energie și va reduce emisiile radiate. Dezactivarea dublorului de ceas se face printr-o simplă schimbare în BIOS, ca în exemplul de mai jos :

1. Deschideți sursa BIOS-ului, fișierul RABBITBIOS.C, aflat în directorul BIOS.
2. Schimbați linia

```
#define CLOCK_DOUBLED 1 // set to 1 to double clock if
                        // Rabbit 2000: crystal <= 12.9024 MHz,
                        // Rabbit 3000: crystal <= 26.7264 MHz,
                        // or to 0 to always disable clock doubler
```

cu cea de mai jos:

```
#define CLOCK_DOUBLED 0
```

3. Salvați fișierul (File -> Save).

2.5.2 Spectrum Spreader

Rabbit 3000 prezintă un Spectrum Spreader care ajută la atenuarea problemelor de interferență electromagnetică (EMI). Spectrum Spreader-ul este în funcțiune în mod implicit, dar poate fi, de asemenea, închis sau setat diferit prin schimbarea următorului macro din BIOS :

```
#define ENABLE_SPREADER 1 // Set to 0 to disable spectrum spreader,
                          // 1 to enable normal spreading, or
                          // 2 to enable strong spreading.
```

Observație: Setarea pentru Spectrum Spreader puternic (valoarea 2) nu este recomandată, pentru că ar putea conduce la limitarea vitezei de ceas sau a ratei de transfer a datelor.

3. Microprocesorul Rabbit 3000

Rabbit 3000 este un procesor modern pe 8 biți, care reprezintă elementul central al unui sistem de dezvoltare embedded complet care cuprinde utilitare de dezvoltare, biblioteci software, module centrale, magazin de componente și suport uman profesionist și mereu disponibil.

Rabbit 3000 împarte setul de instrucțiuni și designul conceptual cu Rabbit 2000. Setul de instrucțiuni este bazat pe Z80/Z180, dar a fost adaptat pentru limbajul C și pentru a permite utilizarea unui MB de spațiu pentru cod.

Rabbit are o gama largă de periferice, incluzând 6 porturi seriale, 56 de pini de porturi paralele de I/O, interfață de control al mișcării, un ceas cu timp/dată, interfață slave și programare în-circuit.

În figura 8 este prezentată schema bloc a microprocesorului Rabbit 3000.

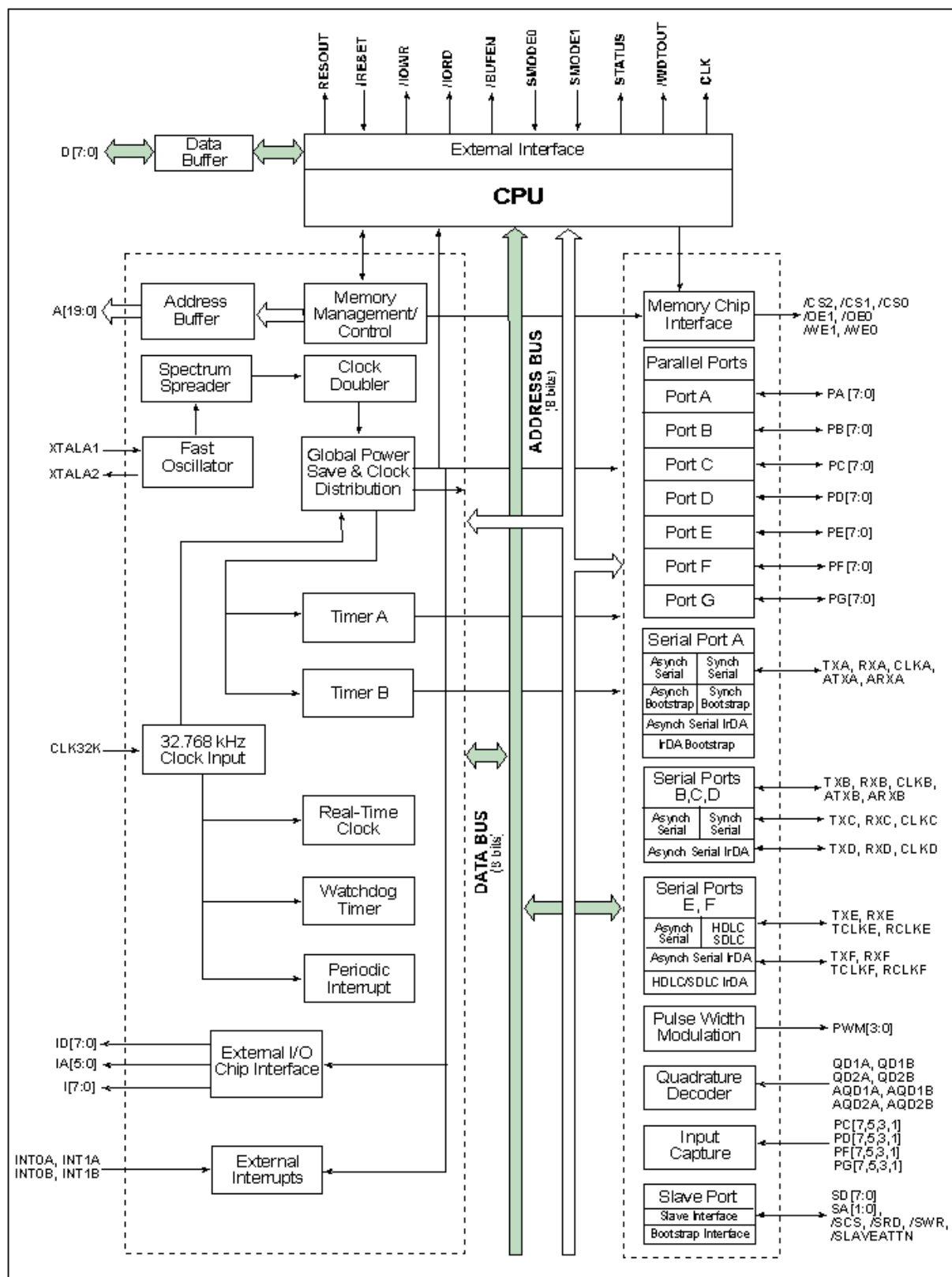


Figura 8. Schema bloc a microprocesorului Rabbit 3000

Rabbit 3000 are un design evoluționar. Procesorul și setul de instrucțiuni sunt aproape identice cu procesorul imediat predecesor, Rabbit 2000. Rabbit 3000 și Rabbit 2000

urmează în mare măsură setul de instrucțiuni și structura regiștrilor de la Z80 și Z180. Comparativ cu Z180, setului de instrucțiuni i s-a adăugat un număr substanțial de instrucțiuni noi. S-a renunțat la unele instrucțiuni Z180 redundante pentru a se putea realiza opcoduri eficiente pe 1 octet pentru instrucțiuni noi și importante. Avantajul acestei abordări evoluționare este că utilizatorii familiarizați cu Z80 sau Z180 pot înțelege imediat limbajul de asamblare Rabbit. Codul sursă Z80 sau Z180 existent poate fi asamblat sau compilat pentru Rabbit cu schimbări minime.

Schimbările tehnologice au făcut unele caracteristici ale familiei Z80/Z180 să fie învechite și s-a renunțat la aceste caracteristici la Rabbit. De exemplu, Rabbit nu are suport special pentru Dynamic RAM, dar suportă extensiv memorie statică. Acest lucru se datorează faptului că prețul memoriei statice a scăzut până la punctul în care a devenit soluția preferată pentru sistemele embedded de scară medie. Rabbit nu suportă DMA (acces direct la memorie) pentru că majoritatea utilizărilor pentru care DMA este tradițional folosit nu se aplică la sistemele embedded, sau pot fi obținute mai bine prin alte metode, cum ar fi rutine rapide de întrerupere, mașini externe de stare sau procesoare slave.

Experiența în scrierea de compilatoare C a evidențiat ineficiența setului de instrucțiuni Z80 pentru executarea limbajului C. Principala problemă este lipsa de instrucțiuni pentru a lucra cu cuvinte pe 16 biți și pentru a accesa date la o adresă calculată, mai ales când aceea dată e pe stivă. Instrucțiunile noi rezolvă aceste probleme.

O altă problemă cu procesoarele pe 8 biți este viteza mică de execuție și lipsa abilității de a putea lucra cu multe operații. O aritmetică bună în virgulă flotantă este o caracteristică foarte importantă pentru sistemele mici. Este ușor de rezolvat multe probleme legate de programare dacă este disponibilă capacitatea adecvată de lucru în virgulă mobilă. Setul de instrucțiuni îmbunătățit al Rabbit asigură o mare viteză de operare cu numere întregi și în virgulă mobilă.

Rabbit suportă patru nivele de priorități ale întreruperilor. Aceasta este o caracteristică importantă care permite utilizarea eficientă de rutine rapide de întrerupere, pentru task-uri de timp real.

3.1. Procesorul Rabbit pe 8 biți în comparație cu alte procesoare

Procesorul Rabbit 3000 a fost realizat cu obiectivul de a crea sisteme practice pentru a rezolva probleme reale ale lumii într-un mod economic. O comparație la prima vedere a Rabbit 3000 cu alte procesoare cu capacități similare ar putea omite anumite puncte forte ale Rabbit.

- Rabbit este un procesor care poate fi utilizat pentru a construi un sistem în care EMI este aproape absent, chiar și la frecvențe de ceas mai mari de 40 MHz. Acest lucru se datorează sursei de curent împărțită, dublatorului de ceas, întinzătorului pentru spectrul de ceas și sfaturilor pentru structura plăcii PC (sau a modulelor de procesor) pe care le oferim. EMI scăzut economisește timp pentru designer care trebuie să întâmpine limite și să treacă teste EMI ale guvernului pentru produsul final.

- Viteza de execuție la Rabbit este de obicei o surpriză plăcută comparativ cu alte procesoare. Aceasta se datorează setului de instrucțiuni bine ales și compact, împreună cu un compilator și bibliotecă excelente. Avem multe benchmark-uri, comparând Rabbit cu familiile de procesoare 186, 386, 8051, Z180 și Z80.
- Magistrala de memorie Rabbit este una foarte eficientă și are un design foarte curat. Nu este necesară nici o logică externă pentru a suporta cipuri de memorie statică. Memoria susținută de baterie este suportată prin funcționalitate încorporată. În timpul operațiilor cu consum redus și ceas redus, ciclul de memorie poate fi redus corespunzător utilizând hardware încorporat, rezultând un consum redus de către memorie.
- Magistrala externă Rabbit utilizează 2 cicluri de ceas pentru ciclurile de citire și 3 cicluri de ceas pentru ciclurile de scriere. Acest lucru are multe avantaje comparativ cu un design single-clock, și după o examinare mai riguroasă, avantajele unui sistem single-clock se dovedesc a fi mai mult imaginate. Avantajele includ: design simplu pentru a evita neînțelegerile între magistrale, cicluri de scriere curate cu timpuri solide de menținere a datelor și adreselor, flexibilitate pentru a avea timpuri de acces la ieșire memoriei mai mari de $\frac{1}{2}$ din ciclul magistralei, și abilitatea de a utiliza un ceas asimetric generat de un dublator de ceas. Presupunul avantaj conform căruia un sistem single-clock are o viteză dublă de magistrală nu este posibil cu memoriile din lumea reală decât dacă memoria este susținută de RAM cu cache rapid.
- Rabbit 3000 operează la 3,6 V sau mai puțin, dar are tolerează 5 V la intrare și are o a doua magistrală pentru operațiile I/O, care este separată de magistrala de memorie. Această magistrală auxiliară poate fi activată de către aplicație ca opțiune de designerului. Aceste caracteristici fac ușoară crearea de sisteme care combină componente pentru 3 V și 5 V și evită problemele de loading și problemele de EMI care rezultă dacă magistrala de memorie este extinsă pentru a se conecta cu multe aparate I/O.
- Rabbit poate fi programat de la distanță, incluzând cold-boot complet, printr-o conexiune serial, Ethernet, sau chiar și printr-o rețea sau Internet utilizând capacitățile încorporate și/sau accesoriul RabbitLink ethernet network. Aceste capacități sunt necostisitoare de implementat.
- Complementul periferic on-chip Rabbit 3000 este foarte mare comparativ cu alte procesoare.

Rabbit este un procesor pe 8 biți cu o magistrală de date externă pe 8 biți și una internă tot pe 8 biți. Deoarece Rabbit utilizează la maxim magistrala externă pe 8 biți și pentru că are un set de instrucțiuni compact, performanțele sale sunt la fel de bune ca multe procesoare pe 16 biți.

Ezităm să comparăm Rabbit cu procesoare pe 32 biți, dar sunt fără îndoială ocazii când utilizatorul poate folosi un Rabbit în loc de un procesor pe 32 biți, astfel economisind bani. Multe instrucțiuni Rabbit sunt de 1 octet lungime. În contrast, lungimea minimă a unei instrucțiuni pe majoritatea procesoarelor RISC de 32 biți este de 32 biți.

3.2. Privire generală asupra perifericelor on-chip și a caracteristicilor acestora

Perifericele on-chip au fost alese pe baza experienței legate de ce periferice sunt cele mai utile la sistemele embedded mici. Perifericele on-chip importante sunt porturile seriale, ceasul sistemului, oscilator oră/dată, porturile I/O paralele, portul slave, codificatoare de mișcare, modulatori pentru lățimea pulsului (PWM), măsurarea pulsului, și timere. Acestea împreună cu alte caracteristici sunt descrise mai jos.

3.2.1. Intrări tolerante la 5 V

Rabbit 3000 operează la tensiune între 1,8 și 3,6 V, dar majoritatea pinilor de intrare tolerează 5 V. Excepție fac pinii de alimentare și pinii pentru buffer-ul oscilatorului. Când un semnal de 5 V este aplicat la pini ce tolerează 5 V, ei prezintă impedanță mare chiar dacă Rabbit nu este alimentat. Toleranța la 5V permite aparatelor ce funcționează la 5V, cu un prag de comutare potrivit, să poată fi direct conectate la Rabbit. Aceasta include componente din familia HCT operate la 5V, care au un prag de intrare între 0,8 și 2 V.

Observație: aparatele CMOS operate la 5V care au un prag la 2,5V nu sunt potrivite pentru o conexiune directă pentru că ieșirile Rabbit nu se ridică mai sus de VDD, care nu poate fi mai mare de 3,6 V și este adesea specificat ca fiind 3,3V. Deși o intrare CMOS cu un prag de 2,5V s-ar putea să comute la 3,3V, va consuma curent excesiv și va comuta încet.

Pentru a transla între 5V și 3,3V, componentele din familia HCT alimentate la 5V pot fi utilizate și sunt adesea cea mai bună soluție. Mai sunt și componente din familia „LVT”, care operează la între 2V și 3,3 V, dar au intrări tolerante la 5V și sunt disponibile de la mulți furnizori. Adevăratele componente de translație dintre nivele sunt disponibile cu pini separați pentru 3,3 și 5V, dar de obicei nu este nevoie de aceste componente și au capcane de design privind secvențierea curentului. Multe cipuri charge pump care fac conversia de tensiune de la DC la DC la preț redus au fost introduse în ultimii ani. Acestea sunt corespunzătoare pentru sistemele cu cerințele duale de voltaj.

3.2.2. Porturile seriale

Sunt șase porturi seriale: A,B,C,D,E și F. Toate acestea pot opera în mod asincron până la o rată de baud egală cu ceasul sistemului divizat prin 8. Porturile asincrone folosesc date pe 7 sau 8 biți, cu sau fără paritate. O schemă de adrese cu un al 9-lea bit, unde un bit adițional este setat sau eliberat pentru a marca primul octet al unui mesaj, este de asemenea suportată.

Software-ul pentru porturile seriale poate să-și dea seama când s-a terminat de transmis ultimul octet dintr-un mesaj de la regiștrii output shift – corectând un defect important al Z180. Acest lucru este important pentru comunicația RS-485, pentru că un half duplex line driver nu poate avea direcția de transmitere inversată până când ultimul bit nu a

fost trimis. La majoritatea UART-urilor, incluzând cele de pe Z180, este dificil să se genereze o întrerupere după ce ultimul bit a fost trimis. Un așa numit bit de adresă poate fi transmis ca high sau ca low după ultimul bit. Bitul de adresă, dacă este utilizat, e urmat de un bit de stop high. Această facilitate poate fi utilizată pentru a transmite 2 biți de stop sau un bit de paritate. Abilitatea de a transmite direct un bit de adresa de nivel de tensiune ridicat nu a fost inclusă în revizia originală a procesorului Rabbit 2000.

Porturile seriale A, B, C și D pot fi operate în modul clocked serial. În acest mod, o linie de ceas clock-ează datele sincron în sau afară. Ceasul poate fi furnizat de portul serial al Rabbit sau de un dispozitiv de la distanță. Când este furnizat de Rabbit, rata baud poate fi până la $\frac{1}{2}$ din frecvența de ceas a sistemului. Când ceasul e furnizat de către un alt dispozitiv, rata maximă de date este frecvența de ceas a sistemului divizată prin 6 din necesitatea de a sincroniza ceasul furnizat extern cu cel intern. Modul serial clock-at poate fi utilizat pentru a suporta dispozitive cu magistrală „SPI”.

Portul serial A are caracteristici speciale. Poate fi utilizat pentru a cold-boot-a sistemul după reset. Acest port este portul normal care este folosit pentru crearea de software sub Dynamic C.

Toate porturile seriale au un mod special de temporizare care suportă standarde de comunicare prin infraroșu.

3.2.3. Ceasul sistem

Oscilatorul principal utilizează un cristal extern cu o frecvență de obicei între 1,8 MHz și 26 MHz. Ceasul procesorului este derivat din ieșirea oscilatorului fie prin dublarea frecvenței, folosind frecvența direct, fie divizând frecvența prin 2, 4, 6, sau 8. Ceasul procesorului poate de asemenea să fie condus de ceasul real-time de 32.768 kHz al oscilatorului pentru operațiuni de foarte mică putere, caz în care oscilatorul principal poate fi închis sub control software.

3.2.4. Intrarea de 32,768 KHz a oscilatorului

Intrarea de 32.768 kHz a oscilatorului este creată să accepte o frecvență de ceas de 32,768 KHz. Un sugerat clock-circuit cu consum redus de energie utilizând componente „de logică mică” este documentat și cu cost redus. Ceasul de 32,768 KHz este utilizat pentru a comanda un numărător pe 48 biți, care poate fi susținut cu baterie (există pin separat pentru alimentare), care servește ca și ceas real-time (RTC). Numărătorul poate fi setat și pregătit prin software și este făcut pentru a păstra ora și data. Există suficienți biți pentru a păstra data pentru mai mult de 100 de ani. Intrarea de 32,768 KHz a oscilatorului este de asemenea utilizată pentru a comanda un cronometru watchdog și pentru a genera ceasul baud pentru portul serial A în timpul secvenței de cold-boot.

3.2.5. Porturi I/O paralele

Există 56 linii I/O paralele divizate între 7 porturi pe 8 biți denumite de la A la G. Majoritatea liniilor pentru porturi au funcții alternative, cum ar fi ca strobe-uri pentru chip select sau serial data. Porturile paralele D, E, F și G au capacitatea de ieșiri sincronizate prin cronometru. Regiștrii de ieșire sunt cascadați ca în figura de mai jos.

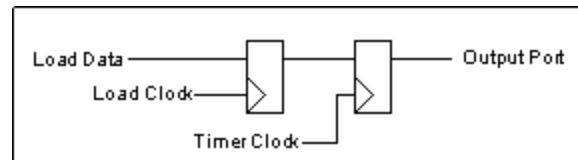


Figura 9. Regiștrii de ieșire cascadați pentru porturile paralele D și E

Stocări către port sunt încărcate în registrul de pe primul nivel. Acel registru este la rândul lui transferat către registrul de ieșire pe un cronometru ceas selectat. Ceasul poate fi selectat să fie ieșire pentru cronometrele A1, B1, B2 sau ceasului periferic (divizat prin 2). Semnalul cronometrului poate cauza o întrerupere care poate fi utilizată pentru a genera pulsuri controlate precis a căror margini sunt poziționate în timp cu acuratețe ridicată. Aplicațiile includ semnalarea comunicațiilor, modularea lății pulsului și motoare driving stepper. (o facilitate separată pentru modularea lății pulsului este inclusă în Rabbit 3000).

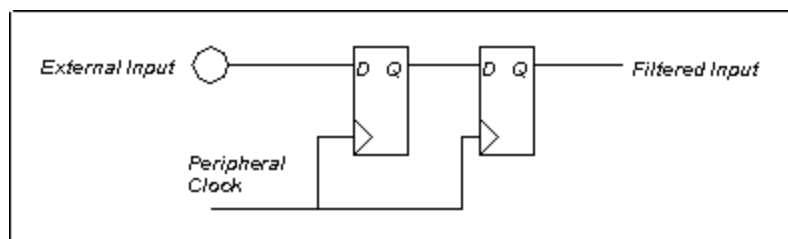


Figura 10. Filtrarea digitală a pinilor de intrare

Pinii de intrare către porturile paralele sunt filtrate de bistabile de tip D cu este arătat în figura 10. Acest lucru împiedică pulsurile mai scurte decât ceasul periferic să fie recunoscute, sincronizează pulsurile externe cu ceasul intern, și evită probleme cu meta stabilitatea (condiții logice temporar nedeterminate datorită timpului de set-up marginal, cu respect pentru ceas).

3.2.6. Portul Slave

Portul slave este creat pentru a permite Rabbit să fie slave pentru alt procesor, care poate fi și un alt Rabbit. Portul este împărțit cu un portul paralel A și este un port de date bidirecțional. Master-ul poate citi oricare din 3 registre selectate prin 2 linii de selectare care formează adresa registrului și un strobe de citire care face conținutul registrului să fie scos prin port. Acești regiștri pot fi scriși ca li regiștri de I/O de către slave-ul Rabbit. 3 registre adiționale transmit date în direcția opusă. Ele sunt scrise de către master cu ajutorul a 2 linii de selectare și a unui strobe de scriere. Fig. 11 arată drumurile datelor prin portul slave.

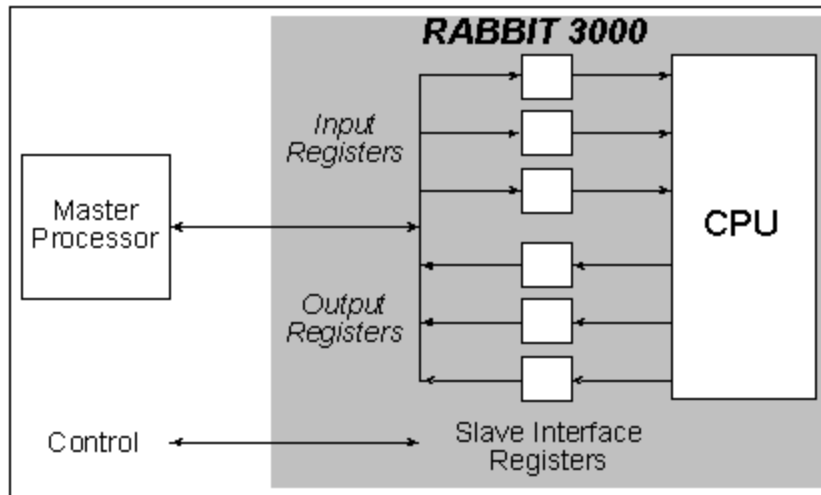


Figura 11. Căile de date pentru portul slave

Rabbit-ul slave poate citi aceiași regiștri ca și regiștri I/O. Când biții de date care intră sunt scriși într-unul din regiștri, biți de status indică care regiștri au fost scriși și o întrerupere opțională poate fi programată să aibă loc odată cu scrierea. Când slave-ul scrie într-unul din regiștri ducând spre ieșire biți de date, o linie de atenționare este activată astfel încât master-ul să poată detecta schimbarea de date și să poată fi întreruptă dacă e necesar. O linie îi spune master-ului că slave-ul a citit toate datele. O altă linie îi spune că alți biți sunt disponibili și nu au fost încă citiți de către master. Portul slave poate fi utilizat pentru a semnala master-ului să execute task-uri utilizând o varietate de protocoale de comunicație deasupra portului slave.

3.2.7. Magistrala auxiliară de I/O

Setul de instrucțiuni pentru Rabbit 3000 suportă acces la memorie și acces la I/O. Accesul la memorie are loc într-un spațiu de memorie de 1 Moctet. Accesul la I/O are loc într-un spațiu de 64 K de spațiu I/O. Într-un design tradițional de micropocesor, aceeași adresă și linii de date sunt utilizate pentru spațiul de memorie și de I/O. Împărțind adresa și liniile de date în această manieră, adeseori forțează compromisuri sau face ca design-ul să fie mult mai complicat. În general magistrala de memorie are mai mult timig critic și mai puțină toleranță pentru încărcare capacitivă adițională impusă prin împărțire ei cu o magistrală I/O.

Cu Rabbit 3000, designerul are opțiunea de activare de magistrale complet separate pentru I/O și memorie. Magistrale de I/O auxiliară utilizează multe din aceiași pini utilizați de către portul slave, astfel operația sa este mutul exclusivă de operația portului slave. Portul paralel A este folosit pentru a furniza 8 linii de date bidirecționale. Pinii de la 2 la 7 ai portului paralel B furnizează 6 linii de adresă, cele mai nesemnificative 6 linii din cele 16 linii care definesc spațiul I/O. Magistrala auxiliară este activă doar pe cicluri de magistrală I/O. Liniile de adresă rămân în aceeași starea de la sfârșitul ultimului ciclu I/O până când un nou ciclu I/O are loc. I/O chip-select și strobe-urile de citire/scriere sunt disponibile și la alți pini astfel

Încât spațiul de 64 octeți definit de cele 6 linii de adresă pot fi ușor expandate. Cilurile I/O se execută și în paralel pe magistrala (de memorie) de bază, când au loc pe magistrala auxiliară, astfel încât linii de adresă adiționale pot fi buffer-ate și furnizate dacă e necesar.

Conectând dispozitive I/O la magistrala auxiliară, rapida magistrală de memorie este scutită de încărcătura capacitivă care altfel ar încetini memoria. Pentru module bazate pe Rabbit 3000, mai puțini pini sunt necesari pentru a ieși din modul pentru că portul slave și magistrala I/O pot să împartă aceiași pini și astfel magistrala de memorie nu mai trebuie să iasă din modul pentru a furniza capacitate I/O. Deoarece magistrala I/O are mai puțină activitate și este mai înceată decât magistrala de memorie, poate fi condusă fizic mai departe fără EMI și probleme de ground bounce. Semnale de 5V pot apărea pe magistrala I/O pentru că intrările Rabbit 3000 tolerează 5V. Semnalele de 5V ar putea cu ușurință cauza probleme pentru magistrala principală dacă sunt conectate memorii de 3,3 V care nu tolerează 5V.

3.2.8. Timerele

Rabbit are mai multe sisteme de cronometrare. Întreruperea periodică este condusă de oscilatorul de 32,768 kHz divizat prin 16, dând o întrerupere la fiecare 488 microsecunde, dacă este activat. Acest lucru este intenționat a fi folosit ca o întrerupere de ceas generală. Cronometrul A constă din 10 countdown-uri pe 8 biți și regiștri de reîncărcare care pot fi cascadați până la o adâncime de 2 nivele. Fiecare registru de countdown poate fi setat să dividă cu orice număr dintre 1 și 256. Ieșirea a 6 timer-e este utilizată pentru a furniza ceasuri baud pentru porturile seriale. Oricare dintre aceste registre poate cauza întreruperi și poate clock-a porturile paralele de ieșire timer-sincronizate. Timer-ul B constă dintr-un numărător pe 10 biți care poate fi citit dar nu și scris. Există 2 registre de potrivire (match registers) pe 10 biți și comparatoare. Dacă registrul de potrivire se potrivește cu numărătorul, un puls este trimis la ieșire. Astfel timer-ul poate fi programat pentru a trimite un puls la ieșire la o numărare predeterminată din viitor. Pulsul poate fi utilizat pentru a clock-a regiștrii de ieșire ai portului paralel timer-sincronizat sau poate să cauzeze întreruperi. Timer-ul B este corespunzător pentru a crea un eveniment la un moment precis din viitor, sub controlul programului.

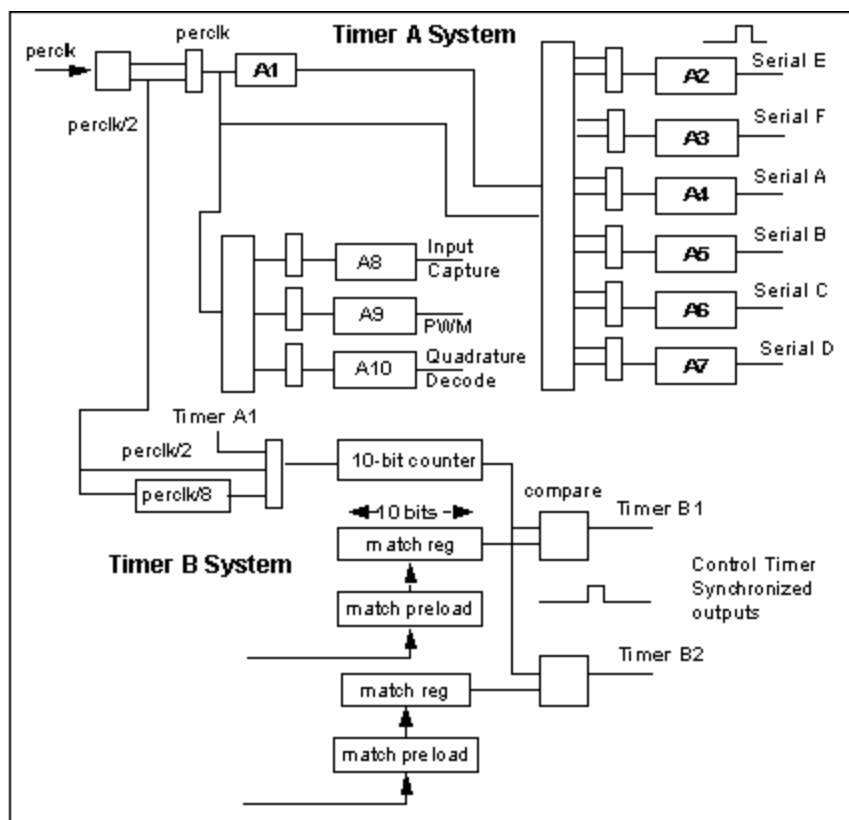


Figura 12. Timerele A și B

3.2.9. Canale de captură la intrare (Input Capture Channels)

Canalele de captură la intrare sunt utilizate pentru a determina timpul la care un eveniment are loc. Un eveniment este semnalat de o pantă crescătoare sau descrescătoare (sau opțional de oricare dintre pante) pe unul din cei 16 pini de intrare care pot fi selectați ca intrare pentru oricare din cele 2 canale. Un numărător pe 16 biți este utilizat pentru a înregistra timpul la care un eveniment are loc. Numărătorul este condus de către ieșirea timer-ului A8 și poate fi setat să numere cu o rată între maxima viteză de ceas și $1/256$ din viteza de ceas.

2 evenimente sunt recunoscute: o *condiție de start* și o *condiție de stop*. Condiția de start poate fi utilizată pentru a începe numărătoarea și condiția de stop pentru a opri numărătoarea. Totuși, numărătorul poate rula continuu sau până când o condiție de stop este întâlnită. Condițiile de start și stop pot de asemenea fi utilizate pentru a memora (latch) timpul curent la momentul când condiția apare decât să pornească și să oprească numărătorul. Aceiași pini pot fi utilizați pentru a detecta condiția de start și de stop, de ex. o pantă crescătoare ar putea fi condiția de start, iar o pantă descrescătoare ar putea fi condiția de stop. Totuși, opțional, condițiile de start și stop pot fi intrări de la pini diferiți.

Canalele de captură la intrare pot fi utilizate pentru a măsura lățimea pulsurilor rapide. Acest lucru este realizat pornind numărătorul la prima pantă a pulsului și capturând valoarea numărătorului pe următoarea pantă a pulsului. În acest caz, eroarea maximă la măsurare este de aproximativ 2 perioade ale ceasului utilizat pentru a număra numărătorul. Dacă este

suficient timp între evenimente pentru a avea loc o întrerupere, unitatea poate fi setată să captureze valoarea numărătorului fie la start fie la stop sau în ambele cazuri și să cazeze o întrerupere de fiecare dată când numărarea este capturată. În acest caz, condițiile de start și de stop își pierd conexiunea cu a opri sau a porni numărătorul și devin doar simple condiții de capturare care pot fi specificate pentru 2 detectoare independente de pantă. Numărătorul poate de asemenea să fie eliberat și pornit sub control software și după aceea să-i fie capturată valoarea ca răspuns la o intrare.

Dacă se dorește, numărătorul de captură poate fi sincronizat cu ieșirile timer-ului B folosite pentru a încărca în mod sincron regiștri de ieșire ale portului paralel. Aceasta face posibilă generarea unui semnal de ieșire sincronizat precis cu un semnal de intrare. De obicei se dorește a sincroniza unul dintre numărătoarele de captură de la intrare cu numărătorul timer-ului B. Offsetul de numărare poate fi măsurat prin trimiterea la ieșire a unui puls la un anumit moment utilizând timer-ul B pentru a seta timpul de output și a captura același puls. Odată ce relația de fază dintre numărătoare este cunoscută, este posibil să se trimită la ieșire pulsuri la un anumit timp după ce s-a capturat pulsul de la intrare, asta dacă delay-ul este suficient de mare pentru ca rutina de întrerupere să proceseze evenimentul de capturare și să seteze pulsul de ieșire sincronizat de timer-ul B. Delay-ul minim de timp necesar este probabil mai mic de 10 microsecunde dacă software-ul este creat cu atenție și viteza de ceas este suficient de mare.

3.2.10. Intrări Codificatoare în cuadratură (Quadrature encoder inputs)

Un codificator în cuadratură este un dispozitiv electromecanic comun utilizat pentru a nota rotația unui ax, sau în unele cazuri să noteze mișcarea unui urmăritor liniar. Aceste dispozitive sunt de obicei implementate prin utilizarea unui disc sau a unei benzi cu benzi alternante opace și transparente care excită detectori optici duali. Semnalele de ieșire sunt dreptunghiulare defazate cu 90° cunoscute de asemenea ca fiind în cuadratură unul față de celălalt. Având semnale în cuadratură, direcția de rotație poate fi detectată notând care semnal conduce celălalt semnal.

Rabbit 3000 are 2 codificatoare în cuadratură. Fiecare unitate are 2 intrări, una fiind intrarea normală și cealaltă intrarea de 90° sau în cuadratură. Un numărător crescător și descrescător pe 8 biți numără pașii decodificatorului înainte și înapoi. Numărarea poate fi extinsă dincolo de 8 biți cu ajutorul unei întreruperi care are loc de fiecare dată când numărarea overflows sau underflows. Semnalele externe sunt sincronizate cu un ceas intern furnizat de ieșirea timer-ului A10.

3.2.11. Ieșiri pentru modularea lățimii pulsului (PWM)

Pulsul cu lățime modulată generează un tren de pulsuri periodice pe o ramă de 1024 cu un duty-cilu care variază între $1/1024$ și $1024/1024$. Există unități PWM independente. Aceste unități sunt conduse de către ieșirea timer-ului A9 care poate fi utilizat pentru a varia lungimea pulsurilor. Când duty-ciclul este mai mare de $1/1024$, pulsurile sunt împărțite în

grupuri distribuite la distanțe de 256 numărări în rama de 1024. Ieșirile modulării lățimii pulsului pot fi trecute printr-un filtru și utilizate ca și convertor D/A pe 10 biți. Ieșirile pot fi de asemenea utilizate pentru a conduce direct dispozitive care au filtrare intrinsecă cum ar fi motoare sau solenoide.

3.2.12. Ceasul de împărțire a spectrului (Spread Spectrum Clock)

Ceasul de sistem principal, care este generat de oscilatorul cristal sau de intrarea unui oscilator extern, poate fi modificată de un ceas de împărțire a spectrului intern al chipului Rabbit 3000. Când împărțitorul de spectru este activat, ceasului i se mărește și descrește viteza în mod alternativ, astfel împărțind spectrul armonicilor de ceas în domeniul de frecvență. Acest lucru reduce EMI și îmbunătățește rezultatele testelor oficiale de emisii-radiate tipic cu 15-20 dB la frecvențe critice. Împărțitorul de frecvențe are 3 moduri de operare: închis, normal și puternic. Timp de acces mai rapid la memorie este necesar când împărțitorul de spectru este utilizat: 2-3 nanosecunde pentru setarea normală, când dublatorul de ceas este activat, 6-9 ns pentru setarea puternică, când dublatorul de ceas este utilizat. Împărțitorul influențează puțin ratele baud și alte temporizări pentru că introduce zgomot (clock jitter), dar este de obicei suficient de mic pentru a fi neglijat.

3.2.13. Pini separați pentru I/O și nucleu

Procesorul Rabbit 3000 este divizat în nucleul logic și inelul I/O. Inelul I/O localizat pe cele 4 margini ale procesorului ține la un loc tranzistorii mari și bonding pads utilizați pentru a crea buffer-ele I/O care conduc semnale către lumea externă. Nucleul, din interiorul inelului I/O, conține procesorul principal logica periferică. Ceasul și marginile ceasului din nucleu sunt foarte rapide cu curenți mari care crează mult zgomot care este comunicat spre exterior prin pinii de alimentare. Buffer-ele I/O au timp de comutare mai încet și operează cel mai des la frecvențe mult mai joase decât nucleul logic. Rabbit are pini separați pentru alimentare și masă pentru nucleu și inelul I/O. Acest lucru permite designerului să alimenteze inelul I/O cu tensiune curată, filtrată pentru a nu mai avea zgomotul generat de către comutarea nucleului. Acest lucru minimizează zgomotul de la frecvențe înalte care altfel ar apărea pe pinii de ieșire conduși de buffer-ele din inelul I/O. Rezultatul este EMI scăzut.

3.3. Standarde de design

Aceeași funcționalitate poate fi adeseori atinsă în mai multe feluri cu Rabbit 3000. Prin publicarea de standarde de design, sau moduri standard de a realiza obiective comune, suportul software și hardware devine mai simplu.

3.3.1. Portul de programare

Rabbit Semiconductor publică o specificație pentru un port standard de programare și furnizează un cablu convertor care poate fi utilizat pentru a conecta un port serial PC la interfața standard de programare. Interfața este implementată utilizând un conector cu 10 pini cu 2 rânduri de pini pe centre de 2 mm. Portul este conectat la portul serial A al Rabbit, la pini pentru modul startup, pe pinul de reset, și la un pin-ieșire programabil care este utilizat pentru a semnaliza PC că atenție este necesară. Cu precauții corespunzătoare în design și în software, este posibil să se folosească un port serial A ca port de programare și ca port serial definit de utilizator, deși acest lucru nu este necesar în majoritatea cazurilor.

Rabbit Semiconductor susține utilizarea portului standard de programare și cablul standard de programare ca port de diagnosticare și de setup pentru a diagnostica probleme sau pentru a seta sisteme în domeniu.

3.3.2. BIOS-ul standard

Rabbit Semiconductor furnizează un BIOS standard pentru Rabbit. BIOS-ul este un program software care gestionează pornirea și închiderea și furnizează servicii de bază pentru software-ul care rulează pe Rabbit.

3.4. Suport pentru Dynamic C pentru Rabbit

Dynamic C este un sistem de dezvoltare interactiv al Z-World bazat pe limbaj C. Dynamic C rulează pe un PC sub sistemul de operare pe 32 biți Windows. Dynamic C furnizează un compilator, editor și debugger. Metoda uzuală pentru debuggarea unui sistem bazat pe Rabbit este să se implementeze conectorul de programare cu 10 pini care se conectează la portul serial PC printr-un cablu convertor standard. Bibliotecile Dynamic C conțin software perfecționat pentru a controla Rabbit. Acestea includ driver-e, utilități și rutine matematice și BIOS-ul pentru debuggare pentru Dynamic C.

În plus, sistemul de operare real-time cunoscut pe plan internațional uC/OS-II, a fost portat pe Rabbit și este disponibil împreună cu Dynamic C cu licență gratuită, pentru a fi utilizat pe produse pe bază de Rabbit.

3.5. Regiștrii microprocesorului

Regiștrii microprocesorului Rabbit 3000 sunt aproape identici cu cei ai microprocesoarelor Z180 și Z80. Figura de mai jos prezintă formatul regiștrilor. Regiștrii XPC și IP sunt noi. Registrul EIR este același cu registrul I al procesorului Z80, și este utilizat pentru a pointera o tabelă cu vectori de întrerupere pentru întreruperile generate extern. Registrul IIR ocupă aceeași poziție logică în cadrul setului de instrucțiuni ca și registrul R de la procesorul Z80, dar funcția sa este aceea de a pointera o tabelă cu vectori de întrerupere pentru întreruperile generate intern.

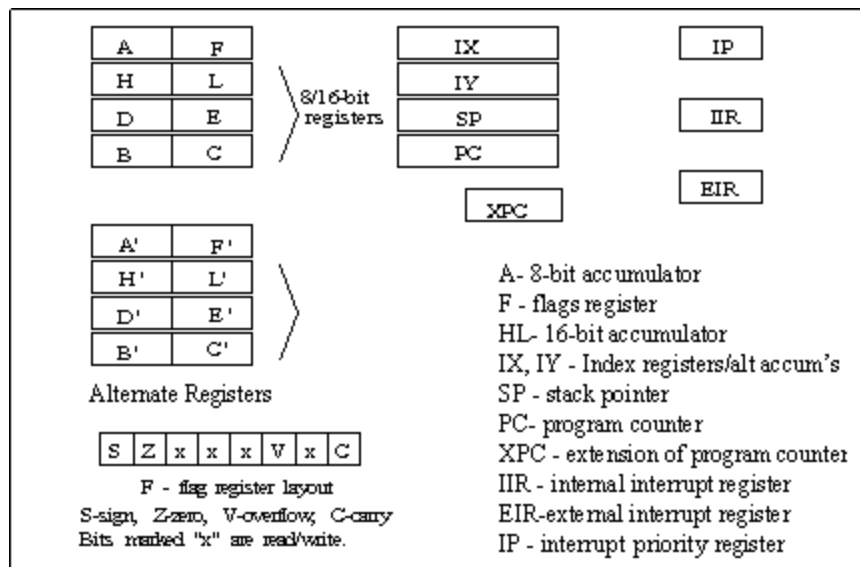


Figura 13. Regiștrii microprocesorului Rabbit

Procesoarele Rabbit (și Z80/Z180) au doi regiștri acumulatori – registrul A servește ca acumulator pe 8 biți pentru operațiile pe 8 biți cum ar fi ADD sau AND. Registrul pe 16 biți HL are rolul de acumulator pentru operațiile pe 16 biți, cum ar fi ADD HL, DE, care adună conținutul registrului pe 16 biți DE la acumulatorul pe 16 biți HL. Pentru multe operații, regiștri IX și IY pot înlocui registrul HL ca și acumulatori.

Registrul F este registrul flagurilor sau registrul de stare. El memorează un număr de flaguri care oferă informații despre ultima operație efectuată. Registrul F nu poate fi accesat direct, decât prin folosirea instrucțiunilor PUSH AF și POP AF. În general, flagurile sunt testate de instrucțiunile de salt condiționat. Flagurile trebuie să indice rezultatele operațiilor aritmetice sau logice în conformitate cu regulile specifice fiecărei instrucțiuni.

Există 4 biți neutilizați în registrul Flagurilor care sunt accesibili utilizatorului prin intermediul instrucțiunilor PUSH AF și POP AF. Acești biți trebuie folosiți cu grijă deoarece noile generații de procesoare Rabbit vor putea utiliza acești biți în scopuri noi.

Regiștrii IX, IY și HL mai pot servi și ca regiștri index. Ei pot pointa adrese de memorie de unde se pot citi sau scrie biți de date. Chiar dacă procesoarele Rabbit pot adresa 1 MB sau mai mult de memorie, regiștrii index pot adresa direct doar 64 KB de memorie (cu excepția anumitor instrucțiuni LDP de adresare extinsă). Spațiul de adrese este expandat prin intermediul hardware-ului de mapare a memoriei și cu ajutorul unor instrucțiuni speciale. Pentru majoritatea aplicațiilor embedded, 64 K de memorie de date (spe deosebire de memoria de cod) sunt suficienți. Procesorul Rabbit poate manipula eficient 1 MB de memorie program.

Registrul SP pointează către stivă, care este folosită pentru legarea subrutinelor și a întreruperilor, precum și ca spațiu de stocare de uz general.

O caracteristică a procesoarelor Rabbit (și a Z80/Z180) o reprezintă setul alternativ de regiștri. Două instrucțiuni speciale fac schimbul între regiștrii alternativi și cei normali. Instrucțiunea EX AF,AF' schimbă conținutul registrului AF cu AF'. Instrucțiunea EXX schimbă

conținutul regiștrilor HL, DE și BC cu HL', DE' și BC'. Comunicația între setul normal de registre și cel alternativ la arhitectura originală Z80 era dificilă, deoarece instrucțiunile de schimb ofereau singura modalitate de comunicare între cele două seturi de regiștri. Procesoarele Rabbit au instrucțiuni noi care îmbunătățesc foarte mult comunicarea între cele două seturi de regiștri. Acest lucru dublează practic numărul regiștrilor disponibili pentru uzul programatorului. Nu este intenționat ca setul alternativ de regiștri să fie utilizat pentru a oferi un set de regiștri separat pentru o rutină de tratare a întreruperii, iar Dynamic C nu suportă acest mod de lucru deoarece utilizează ambele seturi de regiștri liber.

Registrul IP este registrul priorității întreruperilor. Acesta conține 4 câmpuri de câte 2 biți care memorează o istorie a priorităților întreruperilor procesorului. Procesoarele Rabbit suportă patru nivele de prioritate, ceva ce exista doar într-o formă foarte restrictivă la Z80 și Z180.

4. Echipamente și dispozitive folosite

Pentru buna desfășurare a lucrării de laborator se vor folosi următoarele dispozitive și resurse software:

- Modul RCM3365	10
- Placă de bază pentru modulul RCM3365	10
- Sursă de alimentare 12V	10
- Cablu serial pentru programarea RCM3365	10
- Compilator Dynamic C	10

5. Teme

Realizați un referat în care să răspundeți la următoarele întrebări:

1. Care este frecvența de lucru a microprocesorului Rabbit 3000 de pe modulul RCM3365?
2. Câte timer-e sunt disponibile pe modulul RCM3365?
3. Care este dimensiunea memoriei flash? Dar a memoriei pentru date?
4. De câte porturi seriale dispune modulul RCM3365?
5. Cu ce software pot fi programate modulele RCM3365?
6. Care sunt regiștrii pe 8 biți ai microprocesorului Rabbit 3000?
7. Care sunt regiștrii pe 16 biți ai microprocesorului Rabbit?
8. Ce registru are rol de acumulator pentru operațiile pe 8 biți?
9. Ce regiștri pot avea rol de acumulator pentru operațiile pe 16 biți?
10. Ce conține registrul F? Cum putem accesa acest registru în mod direct? Care sunt instrucțiunile care utilizează de regulă acest registru?
11. Care este rolul regiștrilor IX și IY? Ce spațiu de memorie putem adresa cu acești regiștri?

12. Ce spațiu de memorie putem adresa cu un registru pe 8 biți?
13. Care este spațiul maxim de program pe care îl poate utiliza microprocesorul Rabbit?
14. Care este rolul registrului SP? Care sunt cazurile în care se modifică valoarea acestui registru?
15. Ce regiștri conține setul alternativ de regiștri? Cum pot fi utilizați acești regiștri?
16. Care este rolul registrului IP? Câte niveluri de prioritate pentru întreruperi suportă microprocesorul Rabbit?