

Digszar 2018 A.

Csormai Borbála

2019. március 26.

Digszar 2018 A. csoport

0.1. feladat

Adja meg a RISC processzor jellemző tulajdonságait és adjon meg egy példát! Pl: Motorola 88000

- Csak a kívánt alkalmazásra jellemző utasítástípusokat tartalmaz, az utasításkészlet összetettségének csökkentése végett kihagytak olyan utasításokat, amelyeket a program amőgy sem használ, ezáltal nő a sebesség.
- Minimális utasításkészletet és címzési módot, gyors hardver és optimalizált szoftver elemeket használ.
- A komplex függvények elvégzéséhez szubrutinokra, és hosszabb utasítás-sorozatokra van szükség.
- Azonos hosszúságú utasításformátumnak a végrehajtandó utasítás dekódolása minimális idejű lesz.
- Huzalozott utasításdekódolás.
- Egyszeres ciklusvégrhajtás.
- LOAD/STORE memóriaszervezés: 2 művelet - tölt és tárol (regiszter $\leftrightarrow$ memória)

0.2. feladat

Tekintsük a következő paraméterekkel adott 15 bites 2-es komplementes fixpontos rendszert: $p=7$, Mekkora a legkisebb (pozitív) ábrázolható szám? Mekkora a legnagyobb ábrázolható szám? Mekkora a legnegatívabb szám? Mekkora a Δr ebben a rendszerben?

$$V_{FIXEDPOINT} = -b_{N-1} * 2^{N-p-1} + \sum_{i=0}^{N-2} b_i * 2^{i-p}$$

$$p = 7$$

$$N = 15$$

$$\text{legkisebb pozitív szám: } 00000000.0000001 = 2^{-7}$$

$$\text{legnagyobb pozitív szám: } 01111111.1111111 = 128 \quad \text{legnegatívabb szám: } 10000000.0000000 = -128$$

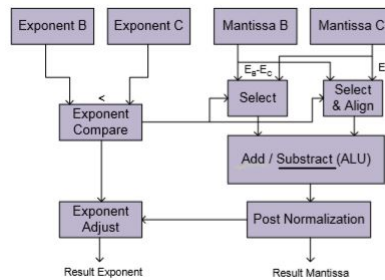
$$\Delta r = 2^{-p} = \frac{1}{128} = 0,0078125$$

0.3. feladat

Adja meg egy 7 bites sorosan írható párhuzamosan olvasható shiftregiszter kapcsolását D tárolók felhasználásával.

0.4. feladat

Adja meg a lebegőpontos kivonó blokkdiagramját, az egyes blokkok funkciójával együtt!

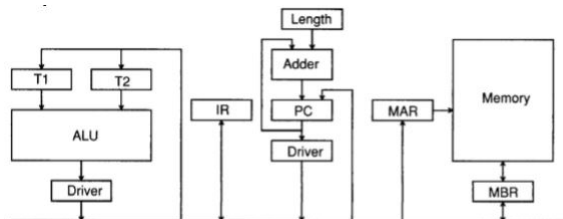


1. ábra.

- a mantisszák hosszát egyeztetni kell, MSB bitek azonos helyiértéken legyenek
- legyen: $0 < B < C \rightarrow C$, vagyis $|EB - EC|$ -vel jobbra shifteljük a kisebb értékű operandus mantisszáját, ami az exponensben jelent változtatást
- SELECT/ALIGN: helyreigazítja a megfelelő mantisszát
- POST NORMALISATION: összeadás/kivonás eredménye egy utólagos normalizálás után a kimenetre kerül

0.5. feladat

Rajzolja fel a kétcímű számítógép blokkdiagramját! Definiálja az egyes blokkok funkcióját is!

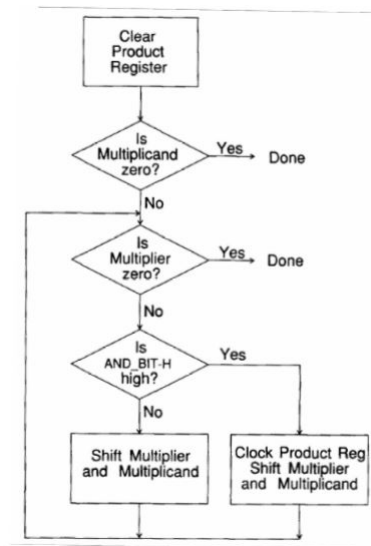


2. ábra. regiszter nélküli címzési mód

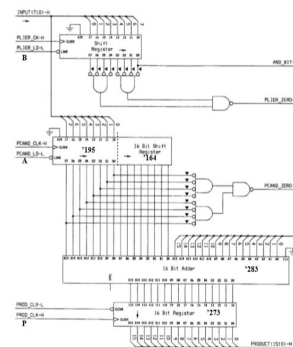
- T1, T2: regiszterek, adattárolók
- ALU: Arithmetic Logic Unit, olyan adatkezelő egység, amely matematikai/logikai műveletek végrehajtását végzi
- Driver: egy adott eszköz elérését, funkcióinak használatát lehetővé tevő szoftver, meghajtó program
- IR: Instruction Register, tárolja az éppen végrehajtás alatt álló utasítást, engedélyezi a gép vezérlő részeinek, hogy a regiszterek, memóriát, aritmetikai egységek vezérlő vonalait a végrehajtáshoz szükséges működési módba állítsák
- PC: Program Counter, A soron következő végrehajtandó utasítás helyét azonosítja.
- MAR: Memory Address Register, a memóriából érlelő vagy a memóriában lévő információ helyét azonosítja adott memóriacím alapján
- MBR: Memory Buffer Register, Tárolja a memóriába bevitt, illetve az onnan érkező információkat
- Memória
- Length
- Adder

0.6. feladat

Adja meg egy 16×16 bites iteratív szorzóalgorithmus folyamatábráját, és ennek áramköri megvalósítását, továbbá az algoritmus végrehajtási idejét! végrehajtási idő:



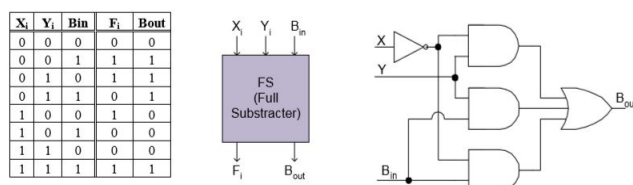
3. ábra.



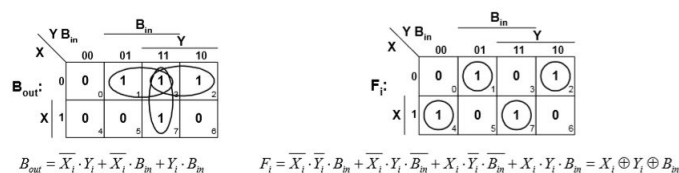
4. ábra.

0.7. feladat

Adja meg egy egybites Full Subtractor (FS) igazságtáblázatát, Karnaugh tábláit, kapusintű kapcsolási rajzát, továbbá a 6-bites RCA (Ripple Carry Adder) kapcsolási rajzát és számítási időszükségletét!

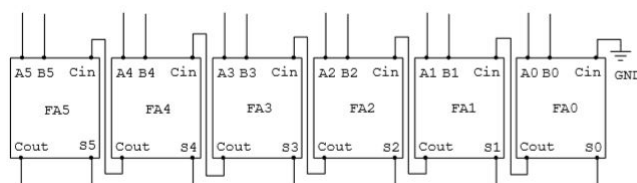


5. ábra.



6. ábra.

számítási időszükséglet: $T_{RCA} = N * T_{RA} = N * (2 * G) = 12G$



7. ábra. Ripple Carry Adder

0.8. feladat

Adja meg a magas-szintű szintézis (HLS) elméleti alaptételét!

Turing Church tézis:

A szimbolikus számítások 3 fő ekvivalens reprezentánsai a következők:

- Turing gép(TM-hardver)
- μ -rekurzív függvény (algoritmusok, program)
- CFG-környezetfüggetlen nyelvtan (OS)

Egy nyelv, amely leírja a teljes tervezési folyamatot, az egyes tervezési lépések jóságát méri: nyelv $\Leftrightarrow$ hardver

0.9. feladat

Adott: memória hozzáférés ideje 20 ns, regiszterből regiszterbe másolás 4 ns, kivonás 6 ns. Adja meg a `ADD2 *X, *Y` RTL leírását és időszükségletét!

Fetch: Regiszterek feltöltése utasításokkal

$PC \rightarrow MAR$ 4 ns

$M[MAR] \rightarrow MBR$ 20 ns

$PC + Ilen \rightarrow PC$ 4 ns

$MBR \rightarrow IR$ 4 ns

Decode: a dekódolást általában 0 idejűnek vesszük

Execute: végrehajtás

$PC \rightarrow MAR$ 4 ns

$PC + XAlen \rightarrow PC$ 4 ns

$M[MAR] \rightarrow MBR$ 20 ns

$MBR \rightarrow MAR$ 4 ns

$M[MAR] \rightarrow MBR$ 20 ns

$MBR \rightarrow MAR$ 4 ns

$M[MAR] \rightarrow MBR$ 20 ns

$MBR \rightarrow T1$ 4 ns

$PC \rightarrow MAR$ 4 ns

$PC + YAlen \rightarrow PC$ 4 ns

$M[MAR] \rightarrow MBR$ 20 ns

$MBR \rightarrow MAR$ 4 ns

$M[MAR] \rightarrow MBR$ 20 ns

$MBR \rightarrow T2$ 4 ns

$T1 + T2 \rightarrow MBR$ 6+4 ns

$MBR \rightarrow M[MAR]$ 20 ns

A teljes (F-D-E) idő: 190 ns

0.10. feladat

Adja meg az alábbi szubrutin CFG (vezérlésfolyamat) gráfját:

0.11. feladat

Adja meg a szekvenciális hálózatok leírási módjait! igazságtábla, algebrai kifejezések, elvi logikai vázlat, állapottábla, állapotgráf