

نحوت سیاست

فصل ۱: اصول زبان های

فصل ۲: نویسنده اطلاعات در پاسیوئر

فصل ۳: اصول برخایه نویسی های زبان اسکیو

فصل ۴: ساختار اصولی پاسیوئر های شفاف

فصل ۵: ترتیبات در ساختار زبان اسکیو

فصل ۶: رسمورات زبان اسکیو پاسیوئر های شفاف

فصل ۷: رسمورات می ساختی و کامپر دان ها

فصل ۸: اسکیو ای رایوس برنامه های اسکیو پسورد COM

فصل ۹: رسمورات لنتل برنامه در رسمورات سیف

فصل ۱۰: نویسنده اطلاعات برخایی پاسیوئر سیز و ران

فصل ۱۱: گرینن اطلاعات از صفحه کلید

فصل ۱۲: عملیات برخایی پاسیوئر اطلاعات در جدول

فصل ۱۳: تولید صوت برخایی با نایکر

فصل ۱۴: وقفه در پاسیوئر های سیف

فصل ۱۵: نظریه سایر روش های آن

فصل ۱۶: مفهوم آن برخایی خانه ها

+ طرز کار پورت صدایی

+ ایکار کار با قایدی

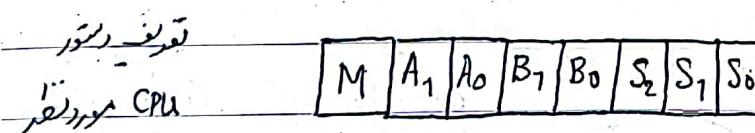
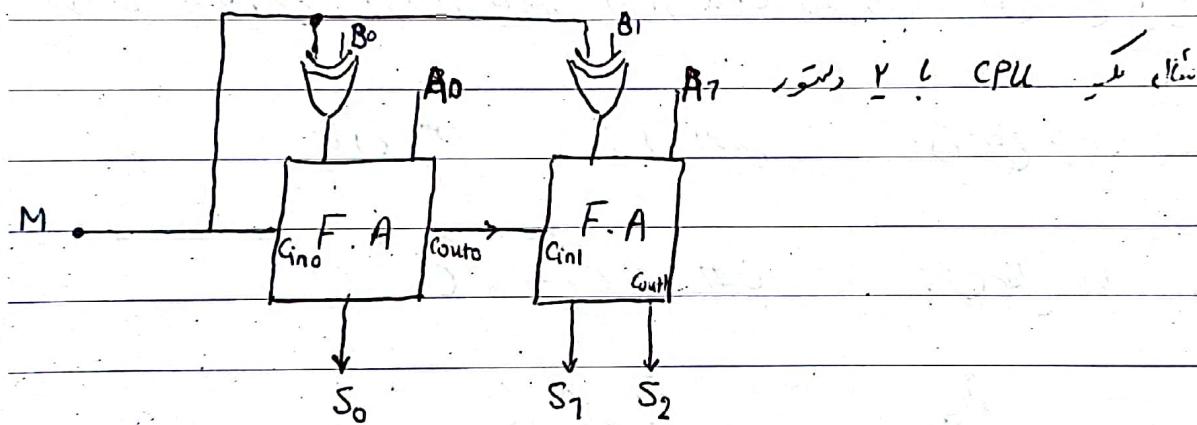
کامپیوئن افزار Emulator 8086

اچوئی رسمورات دیرنامه اسکیو زر صیغه افزار افزار Debug

نام: زبان هنر دانش
نام: رکترین لیدر فنی

۱. مزه بایل زم ۲. مزه بروزه ۳. مزه سایر ۴. مزه ازیزی

زبان هایی: آن دهای است که توسط مایکرور بدل همچ واسطه ای افراد سود و ازدیزی
سری ۵ دستیل شده است



$$M=0 \Rightarrow S = A + B \quad ; \quad M=1 \quad S = A - B$$

نام: ۱. ۰ ۰ ۱ ۰ ۱ S₂ S₁ S₀ S₂ S₁ S₀ = ۰۱ + ۰۱ = ۰۱۰

نام: ۲. ۱ ۱ ۱ ۰ ۱ S₂' S₁' S₀ S₂' S₁' S₀ = ۱۱ - ۰۱ = ۰۱۰

* طبق زبان های مسئل ورده و زنگ ایجاد در برنام نویسی به زبان های کاربر

خطه فرمی است لذا از مسری و دستی بیان کدامی ۱، ۵ (ست) دست سود

کہ رامیڈریٹ بے آن زبان اسمبلی (Assembly language) ہے گوئیں: زبان اسمبلی

لپوور تیغ میں اجرائی تو سما CPU نہیں لذا ایڈیٹر بنا کر اسمبلر Assembler

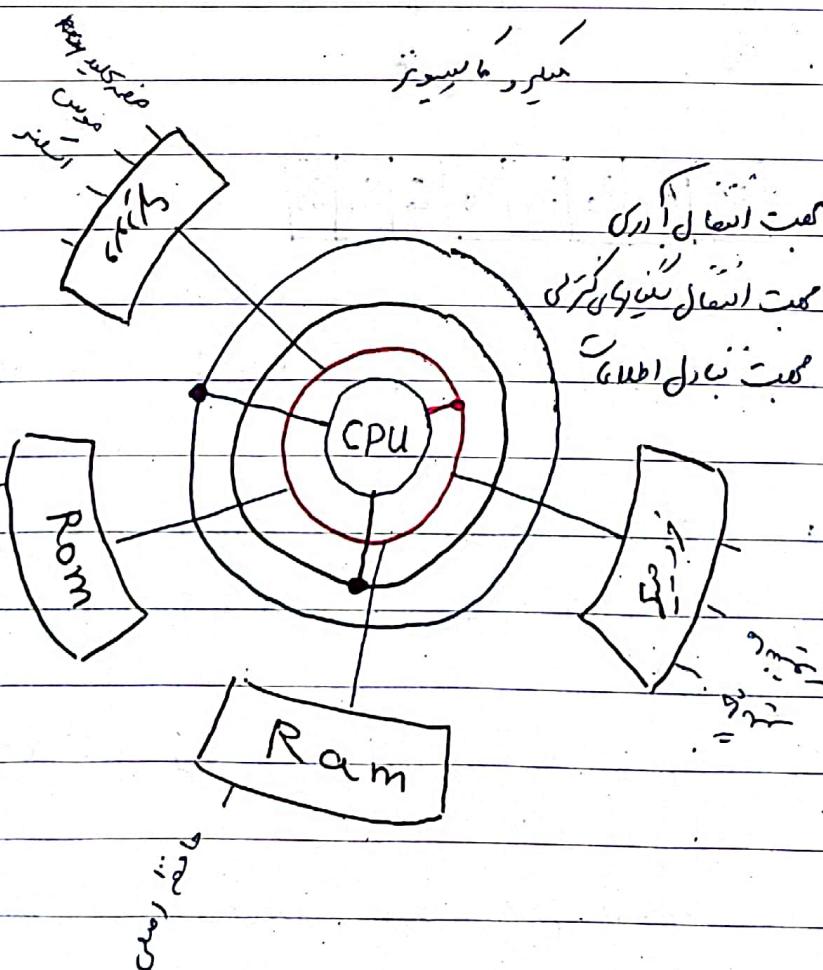
محبت ترجمہ زبان اسمبلی بے زبان یعنی سورہ اس فوادہ میراری لیکر

لہر بے ازایی ہر لکھ زبان ہے۔ باید سہر افزاری در را حل CPU باندھ کے آن کردا

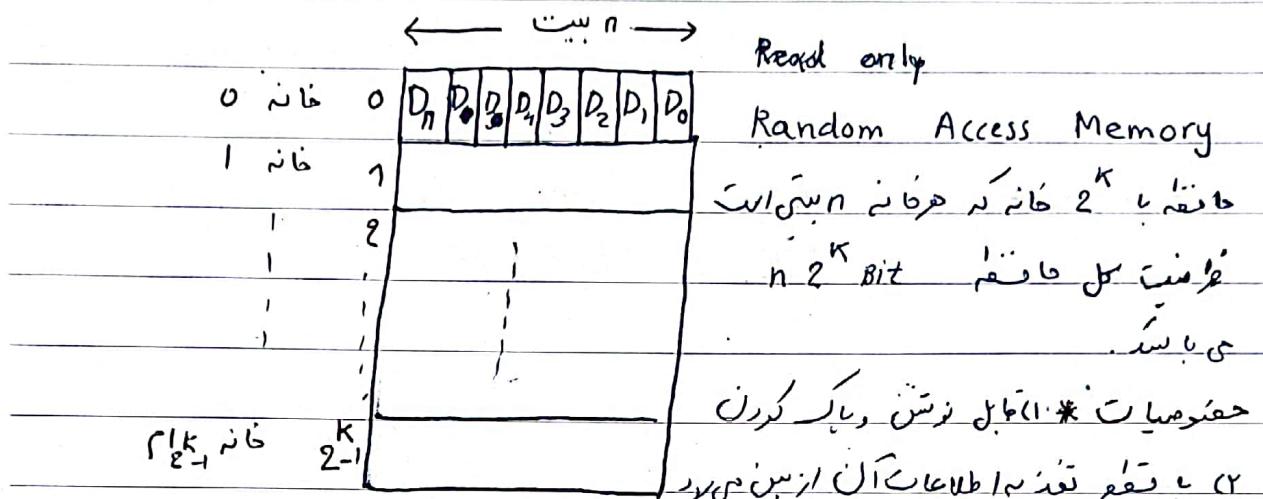
اچھا لکھ بے ایڈیٹر زبان ہے بہت لفڑاری CPU بسٹر کرندے زبان سطح پائیں لوٹیں

زبان، سین، زبان اسمبلی، زبان سطح پائیں

VHDL، C++, C, C, C



حافظه اصلی مایکروپر از سری سلرازی حافظه از جنین شیخ حارس تسلیمی سرور



حافظه عامل نوشت و خواندن

با تغییر تعداد اطلاعات آن ازین میزان بردار

Ram

CPU

با توجه به این سرعت CPU باید لذا جایت بردازی اطلاعات باید اطلاعات

در جایی رُضیه سرور که دسترسی CPU با آن اطلاعات باشود بسیر تر صورت سیر لذا

بدین معنای از Ram استفاده می شود. ولی حمل با تغییر تعداد آن اطلاعات ازین

محدود لذا اطلاعات در (Hard Disk Drive) H.D.D ذخیره می شود

مکانیکی نیز نهاده های سرور کامپیوتر حافظه CPU، حافظه و زدروزی دفوبچی های باید

با این نهاده های دسترسی کوچکی نسبت به هارد کامپیوترها سرعتی داشتند

در برخرازی سی‌ریلیس‌تر رهاییز می‌توان از زبان اس‌بی‌سی ~~میکرو~~ ویا از زبان‌های سطح

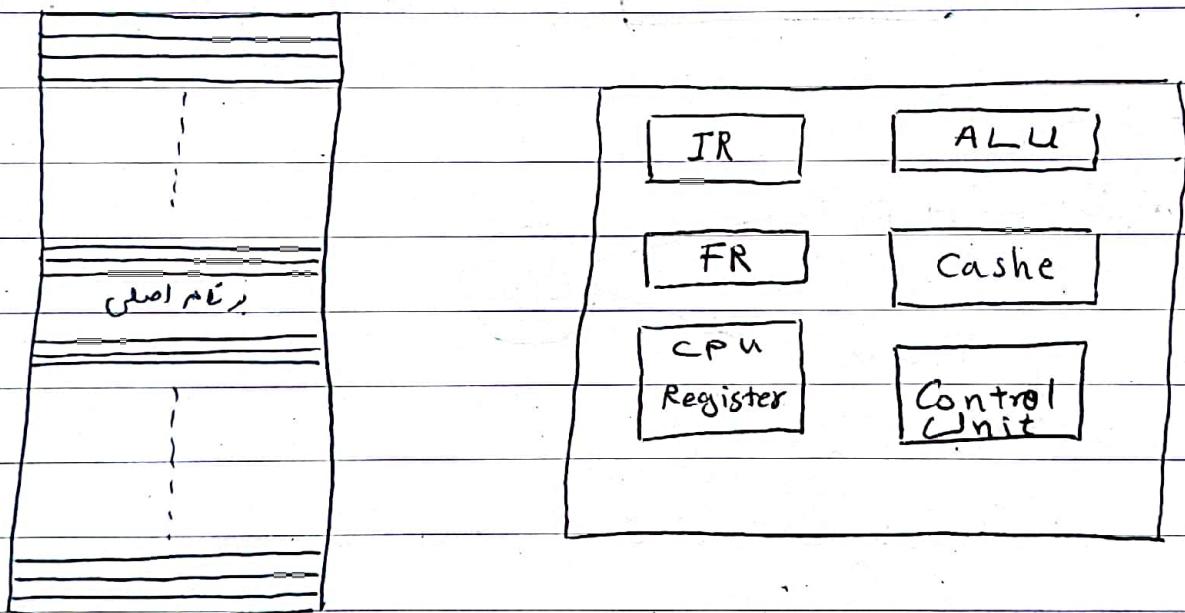
نیازی نمی‌بود $IC_{\text{نمایه}} \rightarrow$ $IC_{\text{نمایه}}$

MC51, 8051, 8052

Microcontroller \rightarrow AVR : Atmega 82

\rightarrow PIC : 16F877

Ram



* باتی برای بسته با ذخیره موقت اطلاعات در بین عبارت برآنها

جهازهای از جانشی سی‌ریلیس‌تر می‌توان DI, SI, DX, CX, BX, AX اسماهند.

رسانهای حسابی و سمعکی در این راه می‌باشد: ALU (Arithmetic and logic Unit)

رسانهای (PC) Program Counter و (IR) Instruction Register *

رسانهای از برنامه مادر که باید ترسیمه CPU (جهازهای در واقع می‌باشند) این راه

مسکونی نه از کدام همان حالت باید که دستور مورثه باید برداشته شود. (۳)

گزینی این پاک دستیع نتیجه لحاظ ریستورات (FR) Flag Register

رسانه صفتی می‌شود آنکه صفت است یا نفع است

Cache : حافظه زنن است که در داخل CPU ساخته هی سرور و حین

در داخل CPU ساخته شده سرور است بسیار بالایی دارد و باعث می شود پردازش اطلاعات

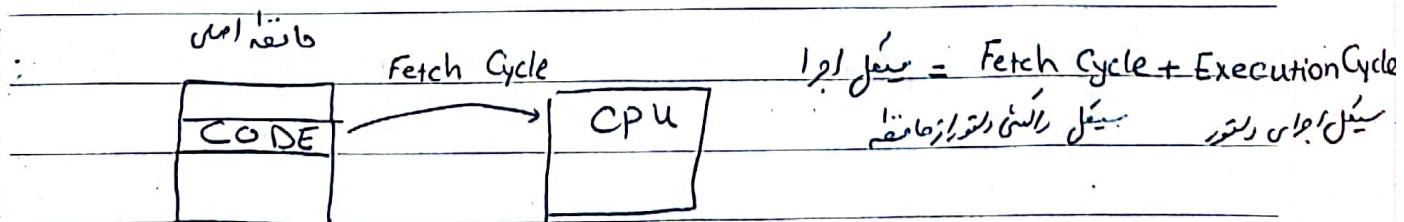
توسیع CPU با سرعت زیاد محدود نیست

کنترل کا مرکزی (C.U.) Control Unit *

دانل CPU است که در اجرای هر دستور درست و مفهوم باید و کار پلیسوند در وقت بازرسی

سینیوی سترل منس این کار را انجام می‌ردد.

سین اجرا و یا زن اجرای دستور: ~~نحوه اجرا و دستور~~ نهادی است که مکرر دستور اجرایی سود.



7	6	5	4	3	2	1	0	نکاره بسته
0	0	0	1	1	0	0	1	
2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	وزن بسته

اعداد بانیری:

در هیات تبلیغ مکرر به ازبسای ۱۰ (ریهال) به مبنای ۲ هست همچنین آن را از تبلیغ مترادی و هست اعدا، آن را از فزب مترادی اسما و همین است.

$$(E1, 4 \wedge V \omega)_{10} = (101001.1011)_2$$

$$\begin{array}{ccccccc}
 & & 2 & & & & \\
 & & \downarrow & & & & \\
 \frac{1}{40} & + & \frac{2}{20} & + & \frac{2}{10} & + & \frac{2}{5} \\
 \textcircled{1} & & \textcircled{2} & & \textcircled{3} & & \textcircled{4} \\
 \frac{2}{20} & & \frac{10}{10} & & \frac{5}{5} & & \frac{2}{2} \\
 \textcircled{5} & & \textcircled{6} & & \textcircled{7} & & \textcircled{8} \\
 \frac{10}{10} & & \frac{5}{5} & & \frac{2}{2} & & \frac{2}{2} \\
 \textcircled{9} & & \textcircled{10} & & \textcircled{11} & & \textcircled{12} \\
 \textcircled{13} & & \textcircled{14} & & \textcircled{15} & & \textcircled{16}
 \end{array}$$

$$\begin{array}{r}
 0.6875 \times 2 = 1.3750 \\
 0.3750 \times 2 = 0.750 \\
 0.1750 \times 2 = 1.500 \\
 0.500 \times 2 = 1.000
 \end{array}$$

$$-1 \rightarrow +91 = 101001 \xrightarrow{\text{مايس بيير}} 0001001 \rightarrow 11010110 \text{ غير منتهي}$$

$$\begin{array}{r}
 11 \\
 23 D 9 \\
 \hline
 8348 \\
 \hline
 A729
 \end{array}
 \quad
 \begin{array}{r}
 413 \\
 59 F \\
 \hline
 2 B 8 \\
 \hline
 2 E 7
 \end{array}
 \quad
 \begin{array}{l}
 \text{dilpi} \\
 F-8=15-8=7 \\
 \text{dilpi} (9+16)-11=14=E
 \end{array}$$

خ سریز : موقعر (Overflow) : عدد بین ۰ تا $2^n - 1$ را بهم قم مسوند و نتیجه هم دیگر $2^n + 1$ نباشد.

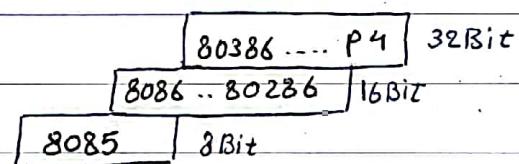
بیتی سواد حی گوئیم نیز ریخ داره است

۴- تشخیص اسبابه توسعه بیت توازن P:

ست توان رم و فرودورهای دینامیکی اضافی می‌شوند که مجموع بیت‌های دنامیکی است.

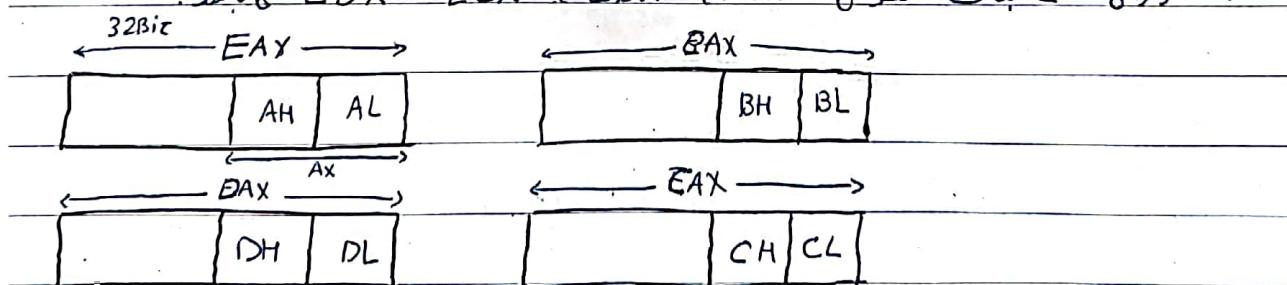
توازن رم و فر

ساده، میکروچارکتی بر IBM-PC



بنات های بررسی

۵- بنات عمومی ۱۶ بیتی CPU



Data Segment (DS); Code Segment (CS) : سیکل

Stack Segment (SS);

بیت بیت های ہر یہم:

۱۰) بیت ہر یہم نظر CF: این بیت ہر یہم سریعیت میں سو رکہ کی بیت نظر از MSB را نہیں بیٹھا.

۱۱) بیت تینیں صفر ZF: زان $ZF = 1$ میں سو رکہ نتیجہ عملیات جیسا کہ مانع فری سو رکہ.

۱۲) بیت ہر یہم خلاست SF، ملاست نتیجہ آفون میں سات راستاں رکھا.

۱۳) بیت ہر یہم نظر کی AF، درجہ سات بالدھی ABCD اگر ایک بیت نظر بیت بعدی ایک بیت نظر

نکس سو رکہ

۱۴) بیت ہر یہم توازن، (PF)

۱۵) بیت ہر یہم سریز OF: این بیت ہر یہم سریعیت مکری سو رکہ نتیجہ عملیات لیکن جیبی بزرگتر از

ستارہ حاصل نہیں سو رکہ ایکارہ گردد

۱۶) بیت فکلرین و قفعہ JF: این بیت بڑی فکل یا نیز فکلرین و قفعہ های کاری اسکے دوسرے کرد.

۱۷) بیت ہر یہم TF: سوچنے کے بیت ہر یہم TF برابر یک اس سے مکر ریکارڈ اس سو رکہ

۱۸) بیت ہر یہم DF: این بیت بڑی نسلی محبت عملیات ریکارڈ ریکارڈ ریکارڈ ریکارڈ

ساخت یا تغییر، بروانه اینمیں ٹائپسیور ہائی سینچ از قسم ہا یا ساخت ہا یا ہئے، درود کو سیلیں کوئی

در ساخت بنا نہ اینمیں روکی تغییر یا کوئی ساخت حقیق قرار ہی کریں، سب ساخت حقیق، مناسب بانیا زما^{۱۶}

بایت (۶۴ کیو بی) حقیق دارو

ساخت دارو: زوره چانی سو رنیا ز در بیان رنیا ساخت قرار ہی کریں، دکرس شروع ساخت در CS قرار دارو

ساخت کو: در ساخت لہ دستورات بروانہ اینمیں دلائل قرار دارو دکرس شروع در CS قرار دارو

ساخت ہستے: در ساخت نسبت صند خانی حقیق برائی ذپریہ اطلاعات دکرس ہائی کہ در اب ایں

بروکہ اینمیں نیز است زروری گردد دکرس اینمیں ساخت سہی تو سہی بیت ساخت نسبت نسبت ۵۵ سینچری کوئی

از آنچا یعنی دکرس شروع در ساخت بر ۱۶ ۷ ۱۰ H قابل تقسیم است لذا طراحان ٹائپسیور جبکہ عربی بروکی

در ساخت اقرار تعمیم گرنے است این صورت دکرس ساخت را کوئی بیکھر اسست در بیان ساخت

ذپریہ کوئی

از نوع دیگر دکرس رضی حقیق

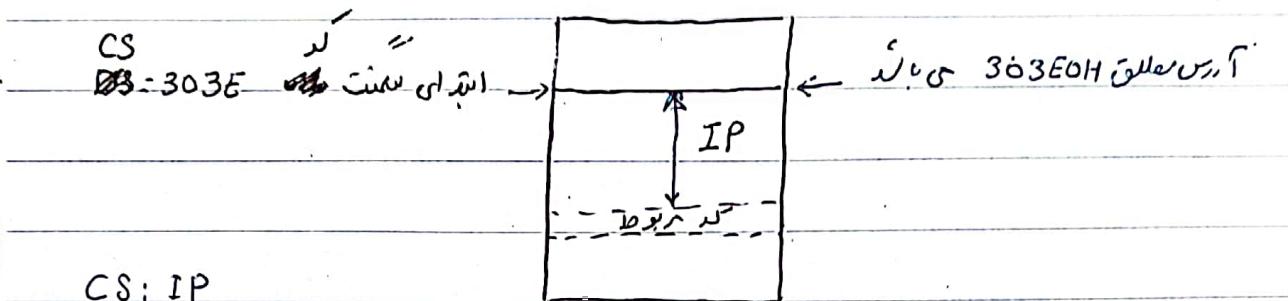
۱) دکرس رہی متعلق: در این دیگر دکرس رقیباً بہ مکانہ سینچ از حقیق اسکرہ ہی نہ

۲) دکرس رضی بہ است

دریک برمجه ایمپلیکت نام گل های ماقمه نسبت به آدرس ایمپلیکت مستقیم یا گردد این بنابراین

اولت چون 12 بیت است لذا می تواند از آدرس FFFFH یا 0000H را آدرس رفع نماید.

حالت اصلی



نحوه خارج کردن بینت نیست، داده، دکدر را فاصله

STAKSG SEGMENT STACK 'STACK'

DW 10 DUP(0) بینت 20 بیت بینان ماقمه بینت

STAKSG ENDS

if SS = 303CH

+ 2H
303EH

باید بینت داده بعد از آدرس 303E کوچک گردد.

البرامیس بینت داده 40 بیت ایکسپلیت نزدیک شود چون

Data1 DB 40 DUP(4)

چون 28 بایت ماقمه در بینت بینی است و 8 بایت بعد از نیز بینت لذا آدرس شروع

DS = 303H

3

3041

بینت نه برابر 3 بینت بعدی بین

توضیح قارگر فتن سنت لینین، داره و کد رهایی

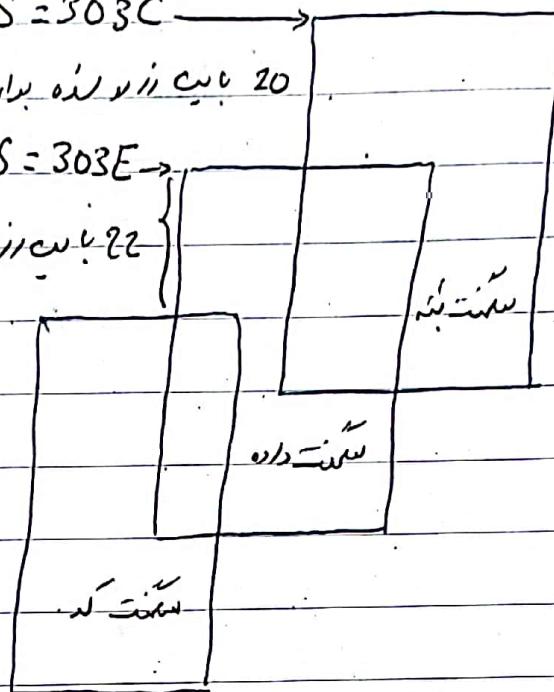
$$SS = 303C$$

20 بین زرده بیان لینین

$$DS = 303E$$

22 بین زرده برای داره

$$CS = 3040$$



حل فتن لینین

با تقدیر Push می توان اطلاعات را در رهایی برگردان اما با استفاده از Pop می توان اطلاعات

از راهی رهایی توان استفاده کرد.

با اجرای دستور Push ابتدا از بارت اسکرین بر لینین SP دو کلید کمی سرور بعد جستجو داره

لینین

مودرن فر در ~~خانه~~ پست ذخیره می سرور بر اجرای دستور Pop می توان از رهایی از رهایی

لینین SP بر آن اسکرین. این در بات تصور دنیا نظر بر رهایی پلیسی خوب SP دو وارد

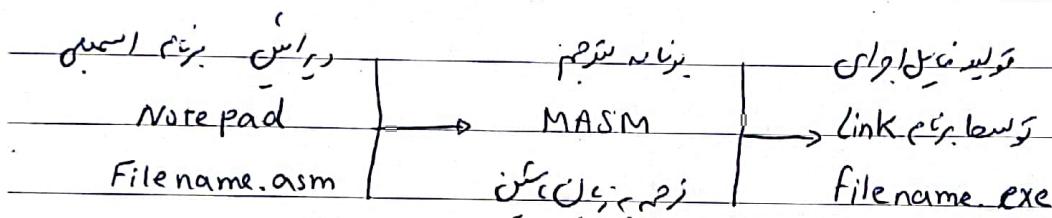
اصفاف می سرور.

AX = 04E17H ; BX = 351AH , SP = 1000H

Push AX SP = OFFEH
Push BX SP = OFFCH
Pop BX SP = OFFEH
Pop AX SP = 1000H

1000	
OFFF	04
OFFE	E1
OFFD	35
OFFC	1A

* ساختار برنامه اسماي



نهائي در نورپايد برنامه اسماي

① برای نوشتن ترجمه در برنامه اسماي از عبارت `تفض` در پایه در ابتدا هر خط برنامه اسماي و در پایه

نوز شتن ترجمه همچنانه را در می شود.

② در مسیر آدرس رکوردر شروع در تابع است که بعد از عدد است و بعدها table

راهنمای از نمایشگر که از قرادرت نمایاری بجهد تجربه شده استواره می شود

main:

برنامه

END , Page , title ④

دستور Page در سریع مکرر برای همین دستور خفوطاً دری می‌میرد دستار مانند کمیم خود را در حفظ

دستور دستور

Page [length], [width] می‌باشد 100,110 را سفین نمی‌نماییم.

برای این که مکرر برای روابط همچنین دستور از دستور Title استفاده نمی‌شود.

Title text [Comment] می‌باشد Title 'Test-Program.Asm' My First Program

END [Entry-Point PROT] نتیجه اینکه برنامه را سفین می‌نماییم

ذکری داشت END باید بر حسب یا نتیجه table سریع دستورات برنامه ایمپلیکت باشد

تعریف ساخت، مکرر برنامه ایمپلیکت در میسرترهای سفین از دستور مکرر است

و مانندیش آرس مکرر با اگر ایست

ساخت که سه دستور تعریف ساخت، ایمپلیکت

برای این دستورات یاراده ها

برای این دستورات دستورات دستورات

لقدیت Procedure ساخت که حاره دسترات بزمیه است مدلل نهایتی یا جند روال.

دارویه با دستور Proc تعریفی من سر شکل هم که دستور Proc بصورت زیر می باشد

نام دوال proc [Attribute] Far
دسترات بزمیه Near
نام دوال ENDP

{ Far
Near }

اس. Subroutine چن Procedure *

اگر دوال از داخل ساخته که در آن تعریف سر کامل فراخوانی باید دوال داخل بوره و با تعریف Near

شی دستوری بصورت رسکی و می توان از نوع Near می باشد. اگر از طرح ساخت در آن فراخوانی هر

از نوع Far خواهد بود.

CodeSeg Segment 'Code'

Main proc

+ باید فراخوانی دلیل دوال از دستور Call اسکریپت

می سر و هب فریم از آن از دستور Ret

بخار می بود بعد از نظری عضو کی خواهد بود.

Main ENDP

CodeSeg ENDS

END Main

دستور ASSUME: ارتباط بین نام هر ساخت ریاضی ساخت را برقرار می کند در آن

که می ناصله قبل از سه دستور Proc قرار می کنند

برای اینها در آن دارویه نهایتی یا جند ساخت می بشد که می خواهد بورت بزمیه

هر ساخت دایر دارد لذا در هر کتف فقط یک از سیستم CPU می تواند دستگاه را در

Assume SS: نام سیستم, DS: نام داده; CS: نام کد

ساخت اسکریپت برای این مدل

Page 110, 100

Title 'Standard_Structure.Asm' Example for an EXE Program

و Define Stack Segment

StackSeg Segment Stack 'Stack'

StackSeg ENDS

و Define data Segment

DataSeg Segment 'Data'

تغیرهای را در اینجا بفرمایم

DataSeg ENDS

و Define Code Segment

CodeSeg Segment 'Code'

Assume SS: StackSeg; DS: DataSeg, CS: CodeSeg

Main proc far این اولین کد نیست که نیاز است این باید در آغاز شود \rightarrow MOV AX, DataSeg
MOV DS, AX

سی دیکسی این کد SS, DS, CS را در اینجا بفرمایم بنابراین این اینجا باید در آغاز شود

Main ENDP

CodeSeg ENDS

END Main

تعريف متغير : بناءً على تعرف متغير من غير ازدواج

و DLR استھا داھی سرور نرم مل ریسٹرات سیدر سرور عملوند Dn

تعریف متغیر Data0 با دایتا تایپ DB و سایز 16

نیز در دریاچه data set میتوان این داده ها را در یک DB در نظر گرفت.

از عبارت زیر '... براي اختصاص رکته‌ی ترانزيستور' Data 2 DB 'String' است.

نموده تعریف نکن این را ۳۰۰۰ کریجی ایجاد کنیم

جیز آرے سا مانگاں رئے کیسے رکارے 'A', 'B', 'C' سور

١٥) Data 5 DB : 10 DUP(0)

داتا 6 DIS 01, 'A', '2000' کے خانے اول راہیں جنگل 01

خانہ درم دالی میٹریس کے اسی A و خانہ درم دالی کے کے اسی برابر 2000
میٹریز کے حافظہ

ترتیب داده از نفع word

Dataword1 DW 1112H	12H	1004
Dataword2 DW	11 H	1005
dw. 16		

* معتاد کو حفظ در آنکس کو حفظ معتاد بزرگتر در همانه کارس بزرگتر

تعریف داره از نوع Quad Word , Ten Byte و Double Word

Data 1 DD 11223344H

• DUP (duplicate) از تکرار نکردن اطلاعات مساده در نتیجه ها از

$$(D1 \ DB \ \overbrace{1,1,1, \dots, 1}^{10}) = (D1 \ DB \ 10 \ DUP(1))$$

ساده سازه سده بزای اسمل

باید این برای های کوچک کنیز نیست خوبی برای های بزرگ داده لیست شمارت این بین

داده سهندی های سهندی سهندی اسماه کردن داشت سهندی باید از این های

اسیدن زیر اسماه کرد

Model Small

با خواست این دستور دستور نیزی بخواست راهنمایی نیست Assume

Stack

تعریف بسته

Data

تعریف داده

Code

تعریف کد

عنوان یا آدرس باید برای سهندی سده اسمل

Page 100, 110

Title 'Sample.asm' a simplified program

Model Small

Stack 64 ; Define stack 32 بایت رزدسته 64 بایت

Data ; Define data segment

تعریف متغیرها یا داده های برای این

Code

Main Proc far

Mov AX, @data ; Set the address Mov AX, dataSeg

Mov DS, AX ; of data Segment to DS. Mov DX, AX

Mov AX, 9COOH ; End of
INT 21H ; Processing
Main ENDP
END Main

تضم المبروز برمجية لغة بحث حار نام سنت سنت

غير معرف بذاته لغة بحث

ri EQU عدد ثوابت : ri تعرف بذاته EQU ثوابت

Count EQU 10 ; D1 EQU 11110000B

org اسارة دلالة لبيانات بعد ار ORG اسارة حانق اسارة

ORG 100H ORG 100H في اسارة حسون
Data1 DB 52 Begin: JMP start

10

دستورات زبان اسماز:

MOV AX, 1234H \rightarrow ADD AL, 0AH آدرس دهنده بیان می‌کند:MOV DS, 2300H (آدرس) \rightarrow هر چنان می‌باید راستگیری را نهاده نشاند تا مقدار دستور از آدرس این:

MOV AX, 2300H

MOV DS, AX

آدرس رهنده شنید.

درین نوع آدرس رهنده در حقیقت می‌باشد و آدرس عدیزد در ریسیور عبارت دارد

Count DB 12H

MOV AL, [4000]

آدرس عدیزد

MOV AL, Count

آدرس سیمبلیک

13 آدرس رهنده غیرتیقیم (SS:BP; DS:DI; DS:SI, DS: BX)

درین روش تقدیم از کتابخانه BP, DI, SI, BX، DS، SS آدرس بیان می‌کرند.

14 آدرس رهنده غیرتیقیم باید باشد، در این روش آدرس برگردان از پردازنده از پایه BX تا SS و DS (درین حدت

پایتیقیم بیان فرضی بیان می‌کند، (BX+BP) 55، (BX+BP) 56، (BX+BP) 57، (BX+BP) 58.

DS: BX

SS : BP

آدرس EA = BP + disp، EA = BX + disp

15/0 MOV CL, [BX]+10 \rightarrow CL باید DS: BX+10 محتوا خانه نشاند.

MOV CL, [BX+10] و MOV CL, 10[BX]

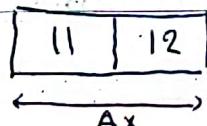
②

MOV BX, 3

MOV AX, [BX+2]

AH AL

0005	12
0006	11
0007	
0008	



۱۵ آدرس رهنگ نمایشیم با بات index (کاربر: تعریف جمل تک بعدی)

درین رهنگ آدرس رهنگ، محتوا را بات index مقدار با جایی disp به عنوان

سیال



آدرس مولت EA در نظر گرفته می شود.

MOV AX, list [DI]

که از اینکو که از اینکو

۱۶ آدرس رهنگ نمایشیم با بات رجیس تری، ای، دی، سی + disp : index

نامه، بات رجیس SI و DI را از عنوان باشم به عنوان پس کوچک کرده، آدرس پس از دی + دی، دی + دی

سیال MOV AX, List [BX] [DI] (کاربر: تعریف مترن ۲ سی)

Page 110, 100

title 'Program1.asm'

StackSg Segment Stack 'Stack'

DW 32H DUP(0)

StackSg ENDS

DataSg Segment 'Data'

DataD DB 5H, 10H, 2H, 15H, OFFH, 32H, 18H, 3H, 7H, 9H

DataSg ENDS

CodeSg Segment 'Code'

Assume SS: StackSg, DS: DataSg, CS: CodeSg

Main Proc Far

MOV AX, DataSg ; که از اینکو

MOV DS, AX ; DS

MOV AL, 78H ; immediate addressing

MOV CL, AL ; Register addressing

ADD AL, DataD ; Direct addressing

MOV AH, data + 1 ; Direct addressing
 LEA DL, data01 ; Move offset address data01 to DL
 MOV CL, [DI] ; Indirect addressing
 MOV BX, offset data01 ; MOV offset address data01 to BX
 MOV CH, [BX] + 2 ; Base relative
 MOV CH, [BX + 3] ; Base relative
 MOV CH, 4[BX] ; Base relative
 MOV SI, offset data01 ;
 MOV DL, [SI] + 5 ; index relative
 MOV DI, offset data + 2
 MOV DH, [DI] + 6
 MOV BX, 5
 MOV DH, [BX][SI] + 2 ; Based index
 MOV DH, [BX][DI] + 3 ;
 MOV DH, [BX + DI + 4] ;
 MOV AX, 4000H ; End of

ست

INT 21H ; Processing

Main ENDP

Codesg ENDS

END Main

بِتَوْبِهِ بِاسْتِئْمَهِ CPU بِأَنْهِيَّ سَهْنَتِ خَاصِّ رَابِّهِ عَنْزَانِي سَيْنِ فَرْضِي مَقْبُولِي هَيْ لَكَنْ حَدَّدَلِ زَيْرِي رَامِي تَوَانِ

لَصِبُورِتِ خَلَاصَاتِي اِزْرَاسِيَّهَايِّ آمِرسِيَّهِ دَيَّبَاتِ سَهْنَتِي سَيْنِ فَرْضِي

بَيَّاتِي سَيْنِ فَرْضِي	آمِرسِيَّهِ دَيَّبَاتِي	آمِرسِيَّهِ دَيَّبَاتِي	آمِرسِيَّهِ دَيَّبَاتِي
immediate	بِعْدِيَّهِ	بِعْدِيَّهِ	نَزَارَهِ $MOV AX, 4507H$
Register	بَيَّاتِهَا	بَيَّاتِهَا	نَزَارَهِ $MOV BX, BX$
Direct	مُسْتَقِيمِي	مُسْتَقِيمِي	مُسْتَقِيمِي $MOV AL, Count^{[999]}$
Register Indirect	بَيَّاتِهَا مُتَعَيِّنِي اِزْفَوْقِي	$[SI], [BX], [DI]$	سَهْنَتِهِ دَارَهِ
		$[BP]$	سَهْنَتِهِ دَارَهِ
base relative	غَيْرِ سَيْتِمِي سَبِّي بَيَّاتِي يَاهِي	$[BX] + disp; [BP] + disp$	سَهْنَتِهِ دَارَهِ
	غَيْرِ سَيْتِمِي سَبِّي بَيَّاتِهِ لِيَنِسِي	$[DI] + disp; [SI] + disp$	سَهْنَتِهِ دَارَهِ دَيَّبَتِهِ
	غَيْرِ سَيْتِمِي بَيَّاتِهَا يَاهِي اِيَنِسِي	$[BX][SI] + disp; [BX][DS] + disp$	سَهْنَتِهِ دَارَهِ
		$[BP][SI] + disp; [BP][DI] + disp$	سَهْنَتِهِ دَارَهِ

بَيَّاتِهَايِّ سَيْنِ فَرْضِي بِرَاهِي بِأَنْهِيَّ سَهْنَتِهَا

بَيَّاتِ سَهْنَتِهِ	CS	DS	ES*	SS
بَيَّاتِي سَيْنِ فَرْضِي	IP	SI, DI, BX	SI, DI, BX	SP, BP
	$\left\{ \begin{array}{l} SI \\ DI \\ BP \\ BX \end{array} \right.$ بِأَنْهِيَّ اِلَّا، وَمِنْ			ES در مُعَدِّياتِ رَسَّتِهِ مُوَدِّعَاتِهِ قَدَّارِيَّهِ تَيَّرِهِ

هَيْ دَسَّرِهِ اِسْتَهَالِهِ آمِرسِيَّهِ : LEA load effective address

آمِرسِيَّهِ بِسَعْيِ اِطْلَاعِي بِمَكِّي بَيَّاتِ سَهْنَتِي سَهْنَرِهِ
ORG 100H 100 101 102 103 104 105 106
DataB DB 23, 46, 12, 90, 51, 4, 30, 40

LEA SI, dataB + 6

MOV AL, [SI]

100H

AL = 30

EA = 100 + 6 = 106



دستور انتقال آدرس (Load data segment Register) LDS

LDS Destination, Source

محل ریسوس به سهنت داده جدید به کار می‌رود

حالت

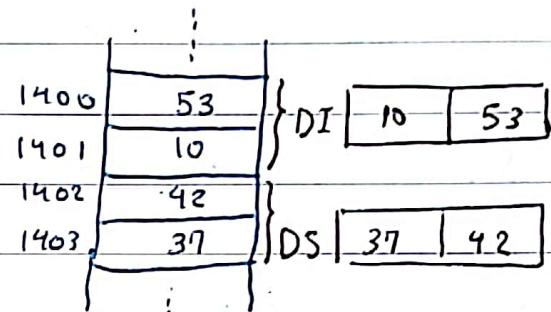
LDS DI, [1400]

DI آدرسی = Source

DI آدرسی + 1 = Source + 1

DI آدرسی + 2 = Source + 2

DI آدرسی + 3 = Source + 3



دستور انتقال آدرس LES یعنی ماتده دستور LDS است فعلاً باید بابت داده DS

LES DI, Source

بایت سهنت داده اضافی اسعاً داده شود

XCHG Operand1, Operand2

دستور

دستور وردی و فرمی: آنی دستورات نکری باید یا نک کنم را هم تواند از دسته داده داده

IN AL, Port بایت AX یا AL دهیم یعنی باید را ایجاد کرد.

IN AX, Port

MOV

OUT Port, AL

MOV DX, 27AH

OUT DX, 378H

OUT Port, AX

IN AL, DX

تیز

OUT DX, AL

تیز

دستورات گامی

ADD Destination, Source ; Destination = Destination + Source ADD (تکمیل کردن)

ADC Destination, Source ; Destination = Destination + Source + CF

SUB dst, Src ; dst = dst - Src

SBW dst, Src ; dst = dst - (Src + CF)

word)

(Convert Byte to Word) ; (Convert Word to CWD), CBW دستورات

Mov AL, +96 ; AL = 01100000

CBW ; AH = 00000000 ; AL = 01100000

Mov AL, -2 ; AL = 1111110

CBW ; AH = 1111111 , AL = 1111110

بسته بودن در بایت ۱۶ دسی را در کامپیوٹر میں کیسے سور

دستور CWD بایت ۱۶ دسی کے بسته بودن بایت AX در بین DX کریں

Mov AX, 0104H ; AX = 0104H

CWD ; DX = 0000H ; AX = 0104H

Mov AX, -32766 ; AX = 8002H

CWD ; DX = FFFFH ; AX = 8002H

MUL Operand

MUL و مزب بایان

اکریباٹ دی کیونا خانقاہ (AX = AL × Operand)

MUL BL ; AX = BL × AL

MUL Byte1 ; AX = Byte1 × AL

در دستور ضرب از این لایه میتوان دستور کل کری باشی باشد که این نتیجه AL در آن ضرب دستور

بایت AX قرار داده میشود و اگر این کلمه باشد نتیجه نتیجه AX در آن ضرب نتیجه کل کمترین کلمه باشد

داده های AX DX و قرار می رسد

Data1 DW 2330H

Data2 DW 3790H

Result DW 2 DUP(?)

MOV AX, Data1

MUL Data2

MOV Result, AX

MOV Result+2, DX

ضرب کری باشد کل کلمه

MOV AL, Byte1

CBW

MUL Word1

دستور IMUL عیناً مانند دستور ضرب MUL است فقط در عمل ضرب علائم هم را تغییر نموده

Byte1 DB 80H

Byte2 DB 90H

Byte3 DB 0F6H

WORD1 DW 8000H

JMUL, MUL

WORD2 DW 2000H

WORD3 DW OFF9CH

WORD4 DW 0020H

WORD5 DW OFF F6H

MOV BL, 20H ; BL = 20H ; Byte 1 = 80H
 MOV AL, Byte 1 ; AX = 1000H
 MUL BL ;
 MOV AX, WORD1 ; AX = 8000H
 MUL WORD2 ; DX:AX = 1000 10000H
 MOV AL, Byte 1 ; AL = 80
 MOV AH, 0 ; AH = 0
 MUL WORD1 ; DX:AX = 0040:0000
 MOV AL, Byte B ; AL = -10
 MOV BL, 10 ; BL = 10
 IMUL BL ; AX = FF9CH = -100H
 MOV AX, WORD3 ; AX = FF9CH = -100
 IMUL WORD5 ; WORD5 = FFF6H = -10 ; DX:AX = 0000:03E8H
 MOV AL, Byte 2 ; AL = 40H
 CBW ;
 IMUL WORD4 ; DX:AX = 0000:0800

in 320x240

X DD 60000	X DD 300000
Y DD 40000	Y DD 250000
Z DD ?	Z DD ?
MOV AX, X	MOV AX, X
ADD AX, Y	MOV SUB AX, Y
MOV Z, AX	MOV Z, AX
MOV AX, X + 2	MOV AX, X + 2
ADC AX, Y + 2	SBB AX, Y + 2
MOV Z + 2, AX	MOV Z + 2, AX

DIV Operand

دستور تقسیم DIV و دستور تقسیم عالیتار IDIV

باید دستور تقسیم DIV، عرور که داخل بیات AX است بر ابرازند یا علودند که بایت که در دستور نوشت

سده تقسیم هی سه دو خارج نسبت در بیات AL، باقیمانده در بیات AH قرار می کنند سه دلیل

علودند در بایت بسیار محترم DX:AX بر آن تقسیم رفع نسبت در AX و باقیمانده در

قماری کسر دارد

* Operand

* دستور تقسیم IDIV نیز عیناً بسیار DIV است فقط مطالعات اعداد را نیز در نظر می کنند

تقسیم کلمه بر بایت

تقسیم کلمه مضافع بر کلمه

MOV AX, 2055 ;

Data1 DD 00105432'

MOV CL, 100 ;

Data2 DW 10000

DIV CL ;

QUOT DW ?

MOV QUOT, AL ;

REMAIN DW ?

MOV REM1, AH ;

MOV AX, WORD PTR Data1

تقسیم کلمه بر کلمه

MOV DX, WORD PTR Data1+2

MOV AX, 10050

DIV DataE

CWD

MOV Quot, AX

MOV BX, 100

MOV REMAIN, DX

DIV BX

MOV QUOT2, AX

MOV REMA1, DX



دستورات ساده: INC, DEC

INC Operand \rightarrow داده
DEC Operand \rightarrow خروجی

Operand = Operand + 1

DEC Operand \rightarrow Operand = Operand - 1

دستور تغییر ملکت: NEG

Packed BCD: 000,0001 تا بیت یک بیت : BCD

Unpacked BCD: 00000000,00000001 تا بیت 8 بیت

لکهای اسلی: اسقال اطلاعات از همه کلید به کامپیوئر با فرستاد کرد اسکن صورت یعنی

لکه اسکن 0 = 30: 01100000

دستورات جمع و تفریق در BCD

محاسبات میانی طیپر همیه به صورت بازیگردی: یا معاشر هنر ارسیل آن می باشد. وی را می شن

در اطلاعات روی ناپتر، را بازیگرد اطلاعات روی خاک، بازیگرد هنر ارسیل استفاده کرد

BCD غیرنیتره تبدیل سود سین با اضافه کردن 30H به آن لکه اسکن تغییر تولید می شود.

محاسبات میانی در نیتره BCD با دستورات SBB و SUB و ADC و ADD در نیتره BCD می شوند. و در نیتره باید از میان دو عدد حاصل اصلاح سود

محاسبات میانی در نیتره BCD با دستورات SBB و SUB و ADC و ADD در نیتره BCD می شوند. و در نیتره باید از میان دو عدد حاصل اصلاح سود

ایجاد سود دیگر باز می شوند. و در نیتره باید از میان دو عدد حاصل اصلاح سود

(٦)

$$\begin{array}{r}
 1001 \\
 0101 \\
 \hline
 1110 \longrightarrow 14 \text{ دهدهم} \longrightarrow \text{دست BCD} \text{ نمی کند} \\
 \hline
 \end{array}
 \quad
 \begin{array}{r}
 1110 \\
 0110 \\
 \hline
 0001 \overset{1}{\text{}} 0100 \\
 \hline
 1 \quad 4 \\
 \text{BCD} \\
 \end{array}
 \quad
 \begin{array}{l}
 \text{بیت عاشر} \\
 \text{AF} = 1
 \end{array}$$

$$\begin{array}{l}
 \text{MOV AL, 18H} \\
 \text{ADD AL, 24H} \\
 \text{DAA} \\
 \hline
 \text{AL} = 42
 \end{array}
 \quad
 \begin{array}{r}
 18H \\
 24H \\
 \hline
 00011000 \\
 00100100 \\
 \hline
 00111100 \\
 0110 \\
 \hline
 01000010
 \end{array}$$

لکته، در قاعده مسند ده نتیجه دستور BCD باشد، ریزیت AL که از BCD نتیجه دستور ADC و ADD باشد، در پایان SBB و SUB نظریه داشتند.

علاریه براین، دستور اصلاح DAS باشد نوشتہ سهوده شده فیصله

Decimal Adjust After Addition: DAA

Decimal Adjust After Subtraction: DAS

حاصل شد BCD صورت

$$(1234)_{BCD} \rightarrow (\quad)_{Hex}$$

تبیین نه BCD به باشندی با حلزنا

$$4 \times 1 = 4 = 4 \times 1H : 4H$$

$$3 \times 10 = 30 = 3 \times 0AH : 1EH$$

$$2 \times 100 = 200 = 2 \times 64H : C8H$$

$$1 \times 1000 = 1000 = 1 \times 3E8H = \underline{3E8H} \\ 04D2H$$

برنامه بنویسید که عدد دو رقم BCD فرد نشاند 62 ریزیت AL را به عدد عادل بگیری

آن تبدیل کاید:

درین فرم نهست راست عدد را در $0AH$ مزب $0AH$ بگیری و با این روش از نهست خاست عدد بگایل کنیم.

1001

(b)

$$\begin{array}{r}
 1001 \\
 0101 \\
 \hline
 1110 \longrightarrow 14 \text{ درجه} \longrightarrow \text{BCD} \text{ نك نك} \text{ بسته} \\
 \hline
 \end{array}
 \quad
 \begin{array}{r}
 1110 \\
 0110 \\
 \hline
 00010100 \\
 1 \quad 4 \\
 \hline
 \text{BCD} \\
 \end{array}
 \quad
 \begin{array}{r}
 AF = 1 \\
 \end{array}$$

$$\begin{array}{l}
 \text{MOV AL, 18H} \\
 \text{ADD AL, 24H} \\
 \text{DAA} \\
 \hline
 \text{AL} = 42
 \end{array}
 \quad
 \begin{array}{r}
 18H \\
 \hline
 24H \\
 \hline
 \end{array}
 \quad
 \begin{array}{r}
 00011000 \\
 00100100 \\
 \hline
 00111100 \\
 0110 \\
 \hline
 01000010
 \end{array}$$

نکته: در قاعده BCD مقداره تابعیم دستور ADC یا ADD با پردازش AL قرار نماید و تغیرات SBB و SUB نماین.

عملیاتی برای BCD با دستور DAS اینها مقدار BCD با پردازش DAA باشد.

Decimal Adjust After Addition: DAB

Decimal Adjust After Subtraction: DAS

بمقدار BCD حاصل نماید

$(1234)_{BCD} \rightarrow ()_{Hex}$

بسط BCD با استری یا چلنزا

$$4 \times 1 = 4 = 4 \times 1H : 9H$$

$$3 \times 10 = 30 = 3 \times 0AH : 1EH$$

$$2 \times 100 = 200 = 2 \times 64H : C8H$$

$$\begin{array}{r}
 1 \times 1000 = 1000 = 1 \times 3E8H = 3E8H \\
 \hline
 04D2H
 \end{array}$$

برای این بثاید که عدد دو تا BCD تردد 62 را در AL باشد عدد معمولی باشی

آن تبدیل کاید!

دوسنی نمی سمت راست عدد را در $0AH$ قریب کنیم و با این روش از سمت راست عدد بیان کنیم.

ANSWER

Page 100, 110

Title 'bcd-bin'.asm

Model Small

.Stack 64

.data

.Code

Main Proc Far

 Mov AX, @data

 Mov DS, AX

 Mov AL, 62H

 Mov BL, AL

 And BL, 0FH

 Mov CL, 4

 And AL, 0E0H

 ROR AL, CL

 Mov CL, 0AH

 Mul CL

 Add AL, BL

 Mov AX, 4C00H

 Int 21H

Main ENDP

END Main

رسو، اصلاح مزب اسن AAM گلزار فرب دلدر اسن دستا بادی اسن اعدار تبدل جمله CDB

نحوه فرب اسکرپت دو نیزه متر دو BCD چهارمین $AL = 05H$ $AL = 0FH$ $AL = 35H$ $AL, 5$ نیزه متر دو BCD چهارمین $AL, '5'$ $Mov AL, '5'$

سکل کوئی سفر 5 مرے کیلئے، 4 مرے کیلئے، 3 مرے کیلئے، 2 مرے کیلئے، 1 مرے کیلئے، 0 مرے کیلئے۔

MUL BL : AX = 0014H
AAM : AX = 0200H → AL = 00 → 0 مربع
AH = 02 → 2 مربع

دستور AAM نعمت موافق اطلاعات در بین AX و CX باشد برای ضرب MUL عمل می‌نماید. دستور AAD نعمت موافق اطلاعات در بین AX و CX باشد برای ضرب AAD ضرب چندین دسیم کد ایسک میل ارائه شود. مزب ایسک فعال بارای تکیر فرم این دستور در لای ضرب چندین رتمن باشد از طرف راست ضرب. بدین قسم نکر این تکمیل می‌نماید فرآوری این دستور میل از این عمل آنست. معمول رانندگ BCN علی قدریه باشد و چیزی در بین AX و CX ایجاد ننماید.

بـسـ تـبـلـ بـلـ هـدـرـ اـعـنـ بـدـ سـبـرـ دـسـتـورـ لـتـيـمـ مـفـسـدـ اـبـرـ عـسـمـ دـلـيـلـ کـهـ درـكـ BCD عـلـيـهـ مـسـرـدـ بـلـ

لشمنه می‌شوند. ورتیت یعنی فارم میت در یکی از AL، بایتیکنیده در یکی از AH و در کنار BCD گلبرگر را

Mon AX, 3435H

Mov DL, '7'

AND DL, OFH

$$DZ \times DL \rightarrow AX = \{ AH = 3, \dots, 10 \}$$

OR AX, 30304

۳۰ ریتر AAD هنر موافقی اطلاعات دریافت \times آندازه رکور DIV عمل نماید

۳) رسته AAD بقعا را می‌رساند مسیریم ریلیات AX در کد BCD نیز فرسته و مسیریم کد عدیتک را از ۰ تا ۹ را کد BCD کلید فرسته و سه عمل آن الله را ایجاد کنید که با یکدیگر

محییت و ریستورات برای لدهار ایم:

درستور که در در ب محورت که اسن را با عزم تغذیه، فرب با عزم می تغذیه به محور می خواهد بود

$$32H = 2 \text{ کروش در } 7 \text{ و } 31H = 2$$

وابد اصلاح آردر نهاد

$$31H + 32H = 63H \text{ عزم می خواهد}$$

دستور اصلاح عزم اسن ریستور

اسن ریستور بعد از دستور عزم نیتیه را بصورت خورا، اصلاح می نماید. حاصل را ب محورت BCD نیز فکر کرده

درستور $MOV AL, 31H$ فتحاباری عزم در اسن کرده

عمل چونه برای محیت اسن خذیری، باید محیت فرن از اسن، همچو رم ماتریس از اسن

$AAA \rightarrow \begin{cases} AL = 8 \\ OR L \end{cases}$ کردن عدد BCD میزد و $30H$ کردن

$$AH = 0$$

تو لیس سود

مثل ۱) که درستور عد ۹ را بکار رفته عد ۵ عزم نیتیه را بصورت کد اسن در آورده

$MOV AL, '9'$

$ADD AL, '5'$

AAA

$OR AX, 3030H$

$MOV AL, 37H$

دستور اصلاح تغذیه اسن

$MOV DH, 32H$

$SUB AL, DH$

AAS

$OR AX, 3030H$



دستورات سندھ بنا سہ دنیا فر:

jmp label

دستورات: JMP

jmp short

surface

CV L. + 12

* جنگ اورس بر صوبہ کے حوالہ

بلاعتر است ۱۲۸- میسرتر

imp Near

—
—
—

دسترسازی JMP فارمادار

مکرس مسکنہ ہدف در داخل سکتیت جنوبی ہیں کلرداروں

jmp Far

میں

د. احمد عاصي

again:

ملحق تماریم برای

دسترات حلقة :

jmp again

~~Mov cx, 10~~

CX حلقة بـ اندازه معتاد 100p ①

• loop again

لئے نہ ہی نہ ہا قصر دیا اور رائے

۴) رسترنیشنزیم : CMP

CMP operand 1, operand 2

اپنی ریاست پرستاری مکتبہ کا ایجاد کرنے والے دوسری بیوی کی بیوی خانم ایک ایسا لذت از اردو

CF ZF

list2 Operand1 → Operand2 : 0 0

Operand1 = 4 0 1

—61— L 11 1:0

دستور `loop2` بازگشاین دستور از سفار، `CX` یک داده‌میز سود و کارائی `00000000` را

است و $z = 1$ است از این میانه دستور `loopz` و `loopc` است و برای دستور `loopnz` و `loopne` است.

جیوئر اریئٹڈ آریس Array میں 100 بیت درج کیا گیا تھا جو دلار جیسے ہائیڈ نے آیا

مکرر ای ۶۳ کاظمی حافظہ مذکور را ان شمار FFH ہستد اور نامہ بستم غار برقرار
لائیں اولین کاظمی ای کو مذکور آئیں برقرار FFH نہیں.

Mov BX, offset array
Mov BX, CX,100

Next: Inc BX

~~REGD~~
GROUP CMP [BX], OFFH
CMP

loopE Next

Mov AX, 4COOH

INT 21H

دستورات پرس سرچی متنی خ

ریس نیگاتیو (jump sign negative) نتیجه میگیرد که باید از مکان فعلی خود را تغییر دهیم.

jNS (jump not Sign n) 语义：

jc ~~ج~~ (jmp if CF=1)

jnc \rightarrow (jump if CF=0.)

jo ~~ws~~ (jump if OF = 1). : jNO

JP / JPE (jump if Parity even)

jnp / jaape jpo (jump parity odd)



Sum \rightarrow 20 . 50 , 150 , 200 , 100 \rightarrow 500

Page 100,110

Title 'add inst.asm'

SSEG Segment Stack 'Stack'

DW 32H DUP(0)

SSEG ENDS

DSEG Segment 'data'

Count EQU 5

Data1 DB 100,200,150,50,20

.ORG 10H

Sum DW ?

DSEG ENDS

CSEG Segment 'Code'

Start Proc. far

assume CS: CSEG, DS: DSEG, SS: SSE6

Mov AX, DSEG

Mov DS, AX

Mov CX, Count

Mov SI, offset data1

Mov AX, 0

Back1: ADD AL, [SI]

JNC over

INC AH

Over: INC SI

DEC CX

JNZ Back1

MOV SUM, AX

MOV AX, 4C00H

INT 21H

Start ENDP

CSE6 EMS

END Start

دستورات پیشگیرانه اندیار عبارت از

JE/JZ (jump equal/or jump if zero) ZF=1

JNE/JNZ (jump not n n if not zero) ZF=0

JG/JNLE (jump greater or jump if not less or equal) SF \oplus OF=0 ZF=0

JGE/JNL (jump greater or equal or jump not less) SF \oplus OF=0

JL/JNLE (jump less or jump not greater or equal) SF \oplus OF=1

JLE/JNG (jump less or equal / jump not greater) SF \oplus OF=1 ZF=1

دستورات پیشگیرانه اندیاری از

JE/JZ (jump equal or jump zero) ZF=1

JNE/JNZ (n not equal n not zero) ZF=0

JA/JNBE (jump above or jump not below or equal) CF=0, ZF=0

JAE/JNB (jump above or equal/or jump not below) CF=0

JB/JNAE (jump Below/or jump not above or equal) CF=1

JBE/JNA (jump Below or equal/or jump not above) ZF=1

دستورات پیشگیرانه از نوع

دستورات پیشگیرانه در صورتی که (A > B) نه A > B نه A = B نه A < B

عنین سه نوع ایجاد می شوند

که اندیار عبارت از

$$AL = 10000110 = 198$$

$$BL = 00010011 = 19$$

$$AL > BL$$

$$AL = -58$$

$$BL = 19$$

$$AL < BL$$

بُنگان بُنگان داده را در حافظه به ترتیب صورتی نمی‌برد. (این در حافظه حافظه)

MOV AL, [BX] (load DI, BX) بُنگان داده

CMP AL, [DI]

jBE DONE

XCHG AL, [DI]

MOV [BX], AL

DONE:

;

(jump if CX is zero) jCXZ (jump if CX is zero)

برای این دستگاه، Max دستگاه، Data1

Page 100, 110

Title maximum.asm 10H

Stacksg Segment para Stack 'stack'

DW 32 DUP(0)

Stacksg ENDS

Datasg Segment para 'data'

Data1 DB 25, 30, 12, 5, 45

ORG 10H

Max DB ?

Datasg ENDS

Codesg Segment para 'Code'

main proc far

assume SS: Stacksg, DS: Datasg, CS: Codesg

Mov Ax, Data1

Mov DS, AX

LEA DI, data1
MOV CX, 4
MOV AL, data1
Back: INC DI
CMP AL, [DI]
JAE SKIP1 jump if \geq to skip1
XCHG AL, [DI]
SKIP1: loop Back:
MOV MX, AL
MOV AX, 4C00H
INT 21H
Main ENDP

CodeSeg ENDS

END Main

دستور رانگریزی دلیل برگشت از دستور Call

دلیل تی Proc دستور Call دلیل از دستور Call است این دلیل برگشت

برگشت

RET

دلیل ENDP

Page 100, 110

Title 'Call_Sub.asm'

.model Small

.Stack 64

.data

.code

Main proc far

Mov AX, @data

Mov DS, AX

Call Sub1

Call Sub2

Mov AX, 4C00H

Int 21H

Main ENDP

Sub1 Proc Near

~~JL Sub2~~

RET

END Sub1 ENDP

Sub2 Proc Near

~~Call Sub3~~

~~JL Sub3~~

RET

Sub2 ENDP

Sub3 Proc

END Main

برنامه بزرگ که مجموع اعداد این

Model Small

Stack 64

10, 15 AL C

Data

First DB 10

Second DB 14

Code

Main proc far

MOV AX, @data

MOV DS, AX

MOV BL, First

MOV CL, Second

MOV AL, 0

Again: ADD AL, BL

INC BL

CMP BL, CL

JLE Again $BL \leq CL$ go to again

MOV AX, 4C00H

INT 21H

main ENDP

END Main

(1)

برنامه بزرگ نیاز ندارد. این دستورات از مجموعه از دستورات آر دی اسی اندیکاتور 30 درجه در

Temperature is too hot درست است. این دستورات این دستورات بقایا است. It is too hot

Model Small

Stack 64

data

Temp_array DB 9, 20, 25, 18, 13, 27, 38, 36, 40, 30

Message1 DB 'Temperature is too hot', '\$'

Message2 DB 'Temperature is suitable', '\$'

Code

Main proc

MOV AX, @data

MOV DS, AX

LEA bp, Temp_array

MOV CX, 10

DEC bp

Again: INC bp

CMP Byte PTR [bp], 30

loop again

LEA dx, Message1

ja EXIT

LEA dx, Message2

exit:

MOV AH, 09

INT 21H

MOV AX, 4C00H

INT 21H

Main endp
end main

دستورات مسحقره ۱

AND Destination, Source $\quad Dest \leftarrow (Dest \wedge) AND (Source) \quad AND$,

OR a, a : OR, 31

XOR 4, 11 : XOR ، ١١

NOT Destination Dest ← (Dest) , NOT ,³⁾

Test Operand1, Operand2 : Test ۱۲۳

بائے ہیں اور دو عملزند بام سنتے سیت AND دوی ستری پرجم (انٹر بلڈنگ کارپوریٹیوں) ہیں جیسا کہ

از عذر نهاده عذر نهاده عذر نهاده

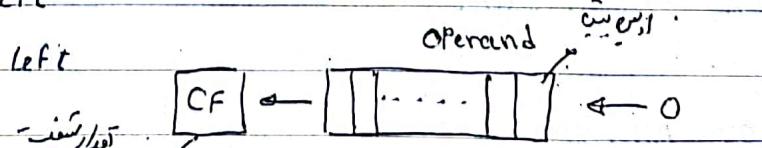
OR DX,DX | OR DX,DX | Test CX,0FFH

jz .. . | js .. . | jz

SHL : shift logical left

SHL : shift logical left

SAL : shift Arithmetic left



is  SAL Operand, N

MOV CL, 3

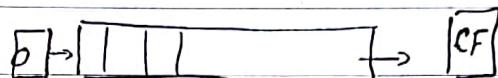
MOV BH, 01010110B

SHL BL, CL

SAL SAL BL, 2

میغایت ب دوست: کسینت شر، SAR، میغایت ب فی، SHR

MOV CL, 02



MOV BH, 10110111B

SHR BL, CL

BL = 00101101

CF = 1

* بیان هندب میگیرد در 2 مه کو ان آن زیارت میگیرد: دادو برای آن تفکیر 2 مه کو ان آن زیارت

SAR



میگایت میگایت خود

MOV BH, 10110111B

SAR BH, 1

BH = 11011011

دیگر ایت خویش

MOV CL, 4

MOV BL, 10110111

Rotate Right (ROR)

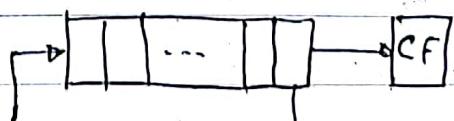
ROR

خویش داشت

ROR BL, CL

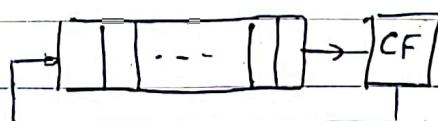
BL = 01111011

CF = 0



Rotate Right through Carry

RCR خویش داشت از دویت بیت همیم



رسور جنیں ہے میں RCR, RCL, ROL، ROR میں درجت صیغہ میں لیا جائے۔

برام بیٹھ بکہ ایکریت ۲ تھیں دیکھیں Data1 تک بور جو ایک BX AX تک داد ایسا فی سردار، اور

• Model Small

• Stack 64

• Data

Data1 DB \$11101113

• Code

~~Motor example data~~

Begin Proc

Mov AX, @data

Mov DS, AX

Mov AL, Data1

ROR AL, 3

JC Carry

Inc DX

Jmp Exit

Carry: Inc BX

Exit: Mov AX, 4C00H

INT 21

Begin

Stack

ENDP

END Begin

برنامه ای بنویسید که تعداد ۱ موجود در سیستم آنرا در یک سیورت نماید

Model Small

Stack 64

Code - Data

AA DB 10110111H

Code

Begin Proc

MOV AX, @data

MOV DS, AX

MOV BL, 1

MOV CL, 8

Again: ROR AA, 1

JNC Next

INC BL

Next:

Loop again

MOV AH, 02H

MOV DL, BL

ADD DL, 30H

INT 21H

MOV AX, 4C00H

INT 21H

Main ENDP

END Main

از آن ب بعد نویسی هر این اندیشه ها را کامپیوتر های متغیر نویسید. نصفها... خلیج ر.

دسترسی ب پرورشی کامپیوتر از طبقه آرسی بورت اول بینهایت. شروع بورت پرورشی بعد از 000H

الی 3FFH متوالند از آرسی آرسی های نزدیک آرسی 00H ال OFFH چهار بورشی دارند کامپیوتر

تصییر بورت صفحه کلید، بلندکار، ناپر---من بندکه بحیره دفعه آرسی های دفعه از 100H ال 3FFH برای بینهایت

کامپیوتر نظر بورت سریال، بورت پردازش A/D و ... دارد

برای ارتباط با دستگاه هایی مانند آذوقه از دستورات OUT، IN،

دستگاهی رفعه های DOS، BIOS، پرینتر دسترسی دارد.

نمایه آرسی بینهایت از دسترسی کامپیوتر

نوع بورت

نمایه آرسی بورت

DMA لذتمند

000H - 021H

لذتمند اسکرین دفعه بینهایت

0020H - 0021H

بورت صفحه کلید (فراشندر از پرینتر، کلید فریزه ای)

060H

بورت بلندگو

081H

بورت پارالل (LPT1)

378H - 37FH

آخر لذتمند دستگاه

3F0H - 3F5H

بورت سریال

3F8H - 3FFH

IN AL, Port_Ad , IN AX, Port_Ad : IN ریتر

ریتر IN : این ریتر که باید ریکلکم از پرینت آدرس آن بوسی DX میگیرد سند این

کاربرایت میگیرد که آدرس آن بوسی ES:DI میگیرد. قارچ راهد سی از اسکال

بها فتح برای ریتر باید ملک واحد و برای ریتر کلمه ای در داده برجیز DI افغانی خواهد

MOV CX,20

MOV DX,379H ES:DI ۲ بایت باز پرینت در پرینت میگیرد

LEA DI, INPSTR آن اکاره میگیرد قارچ راهد

REP INSB تا زمانی CX صفحه کنور تکرار ادامه دارد

OUT Port, AL , OUT Port, AX : OUT ریتر

ریتر OUT : نماینده بایکلم راهه برخاهم بسی ES:SI و ملک واحد دارای ریتر اند

جیف سند توسعه DX قارچ ایند سی از اسکال باید ریتر که باید ملک واحد و برای ریتر اند

MOV CX,15

MOV DX,378H کلمه ای در واحد برجیز SI افغانی ریتر

LEA SI, OUTSTR

Rep OUTSB ۱۵ بایت سی از اسکال بسی ES:SI ملک واحد

خواهد بود با آدرس سیف سند توسعه DX ارسال گردد.

برای ریتر باید ملک از پرینت INSB درباری اسکال کلمه ای ملک از پرینت از پرینت