**הארכיטקטורה של ה-x86\_64**

**מעבדי המחשבים האישיים היום עובדים היום (במחשבים חדשים בדרך כלל) במוד 64 ביט.**

**זה אמור שהמערכת ההפעלה וחלק מהיישומים רצים במוד 64 ביט, היישומים שקומפלו עבור מודל התוכניתן 32 רצים במוד 32 ביט הנועד לתאימות אחורה compatibility mode.**

**יישומים 32 ביט רצים כמעט באותה מהירות כמו במערכת 32 ביט.**

**במובן מסוים המעבד תומך בשני מודלי תוכניתן.**

**ב-long mode: 16 אוגרים בגודל 16 ביט, כולם יכולים לשמש כפוינטרים.**

**המוטיבציה העיקרית של הארכיטקטורה החדשה:**

**הגדלת מרחב הכתובות הוירטואלי והפיזי.**

**מרחב הכתובות הלינארי ניתן להרחבה עד 264 = 1.8x1019**

**מרחב הזיכרון הפיזי/כמות הזיכרון הפיזי הניתן להרחבה עד 252 = 4.5x1015 (בערך 4000 טרה).**

**בפועל המעבדים הנוכחיים לא תומכים בזה.**

**במעבד המקורי Operton 2008 עד 248 מרחב כתובות וירטואלי, עד 240  מרחב כתובות פיזי.**

**בשעה זו:**

**עד 2.8x1014= 248 מרחב כתובות וירטואלי, ואותו דבר עד 248  מרחב כתובות פיזי.**

**הדבר נעשה על מנת לחסוך מימוש טבלאות שאין צורך בהם.**

**מרחב הכתובות הפיזי החוקי מ-0 עד גודל הזיכרון פחות 1**

**מרחב הכתובות הלינארי החוקי:**

**כל הכתובות החוקיות כיום נקראות כתובות קנוניות:**

**כל הכתובות מהצורה**

**0 46 47 48 63**

|  |  |  |
| --- | --- | --- |
| **0 …..............0** | **0** | **XXX…………………………………XX** |

**או**

**0 46 47 48 63**

|  |  |  |
| --- | --- | --- |
| **1 …..............1** | **1** | **XXX…………………………………XX** |

**באופן ציורי:**



**אוגרים מיוחדים:**

**אוגרים CR0, CR2, CR3 הורחבו ל-64 ביט.**

**אוגרים GDTR, IDTR הורחבו ל-80 ביט.**

**אוגר LDTRעכשיו 96 ביט (עם ה-selector 112 ביט)**

**אוגר חדש CR4**

**אוגר חדש CR8**

**אוגר הדגלים RFLAGS עכשיו 64 ביט.**

**קבוצה חדשה של אוגרים מיוחדים Model Specific Registers במסגרת לא מחייבת שינוי בשפת המכונה אלא דרך 2 פקודות מכונה מיוחסות RDMSR,WRMSR**

**אוגרי הסגמנט והסלקטורים עדיין 16 ביט.**

**אוגרי הסגמנטים:**

* **CS,DS,SS,ES אינם מספקים Seg Base, Seg Limit יותר.**
* **GS,FS מספקים רק Seg Base.**
* **DS, SS, ES חדלו כמעט לחלוטין לתפקד.**
* **CS מספק עכשיו רק DPL, C, D, ו-L. L ביט חדש.**

**המקום שבו נקבע אם קוד מורץ ב-long mode או compabilty mode הוא ביט ה-L.**

**חישוב כתובות לינאריות ב- x86\_64 long mode**

**על פי רוב הכתובת הלינאריות יהיה זהה לכתובת האפקטיבית.**

**הכתובת האפקטיבית – תוכן הסוגריים המרובעות.**

**החריגים היחידים הם אם משתמשים ב-segment override עם האוגרים FS,GS. אז מסכמים גם את ה-FS.BASE או GS.BASE.**

**כלומר:**

**Linear Address = Effective Address**

**Linear Address = FS.base +Effective Address**

**Linear Address = GS.base + Effective Address**