سجل الإقلاع الرئيسي

MASTER BOOT RECORD

سجل الإقلاع الموجود في القطاع 0 على الوسيط المقسم، في منصات x86، أنظمة BIOS و/أو جزئيا في UEFI/GPT (منذ 2014)

نسخة 2

يناير/كانون الثاني 2019

سجل الإقلاع الرئيسي

عبارة عن قطاع إقلاع [16] على قرص مقسم، يحتل القطاع المطلق 0 (عند الكتلة رقم 0 بحساب التدوين الخطي المسمى عنونة الكتل المنطقية LBA أو بالتدوين الثلاثي الفيزيائي عند العنوان (CHS 0,0,1) [31] حجم هذا القطاع التقليدي 512 بايت، (بينما في قراص AF. و SSD حجم القطاع غالبا سيكون 4 كيلوبايت [48]).

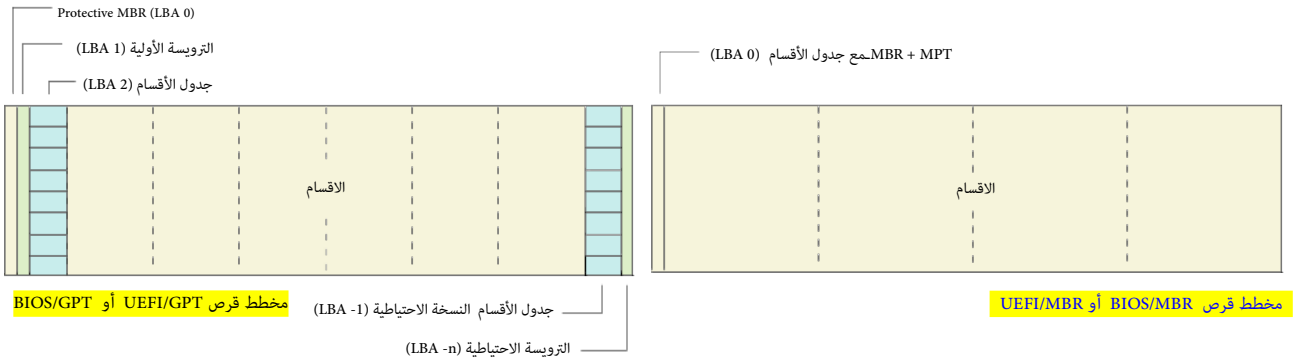
هذا القطاع ينشئ آليا عند تقسيم الوسيط في بداية أجهزة التخزين الكبيرة، مثل الأقراص الثابتة والأقراص القابلة للإزالة المستخدمة في الأنظمة المتوافقة مع أنظمة IBM وأنظمة أخرى.

سجل الإقلاع الرئيسي يحتفظ بمعلومات تصف تنظيم الأقسام المنطقية المتضمنة في العادة أنظمة ملفات على القرص، إلى جانب شفرة تنفيذية [21] [18] لتحميل نظام التشغيل، الشفرة في الغالب تدعى محمل إقلاع [1] [44]، عادة، هذه الشفرة تقوم باستدعاء المرحلة الثانية من شفرة الإقلاع، أو ترتبط بشفرة سجل إقلاع القسم VBR. مثلا، في معظم توزيعات لينكس، المرحلة الثانية من شفرة محمل الإقلاع GNU GRUB 2 يمكن أن تقع في القطاعات التي تأتي بعد MBR أو حتى داخل نظام الملفات.

512 بايت في بداية المسار الأول (المسار 0) تتضمن سجل الاقلاع – على القرص الثابت

سجل الإقلاع الرئيسي لا يستخدم في أقراص أكبر من 2 تيرابايت ($2^{32} \times 512$ بايت) [2] لأن تنظيم جدول الأقسام MPT مقيد بمساحة التخزين القابلة للعنونة [46] والتي لا يمكن أن تتجاوز 2 تيرابايت على أقراص MBR والطرق التي تحاول رفع هذا الحد جزئيا بالأخذ بحساب 33-بت أو قطاعات 4096-بايت، ليست مدعومة رسميا، لأنها تكسر التوافق الذي بين محملات الإقلاع الحالية ومعظم أنظمة التشغيل والأدوات المتوافقة مع MBR، ويمكن أن تسبب تلف حقيقي في البيانات عند استخدمها خارج بيئات أنظمتها المتحكم بالكاد فيها. لهذه الأسباب ولأسباب أخرى، منذ عام 2010 بدأ التحول إلى استعمال GPT (الذي هو جزء من مواصفة UEFI)، أي أن تخطيط GPT سيكون إجباري في الأقراص الأكبر من 2 تيرابايت. ويحتاج إلى إنشاء قسم إقلاع خاص في أقراص UEFI/GPT، يسمى اختصارا ESP. (نوع EFh) (لمعلومات أكثر راجع: كتيب "جدول أقسام GUID" باللغة العربية أو راجع مواصفة UEFI باللغة الانجليزية)[43].

في الواقع، يمكن أيضا استخدام تخطيط GPT في أقراص BIOS [20]، ويسمى هذا غالبا إقلاع BIOS/GPT لكن بشرط إنشاء قسم إقلاع BBP، مع تنصيب GRUB 2 في ذلك القسم، في هذه الحالة GPT سيتضمن أيضا نسخة هجينة من MBR تدعى اصطلاحا protective MBR، (نوع EEh) تستخدم للتوافق مع الإصدارات السابقة (تمنع أنظمة BIOS/MBR من تعديل أو حذف GPT).



سجل MBR لا يوجد في وسائط التخزين التي بدون أقسام مثل القرص المرن، أو تنويعات قرص [61] superfloppies. أو ما شبه ذلك (لأن تلك الوسائط تملك قطاع إقلاع خاص يدعى VBR).

في أنظمة BIOS/MBR عند إقلاع الجهاز، [18] [20] البرنامج الثابت BIOS سيكون المسؤول عن تحميل وتشغيل MBR، بعد تفحص العتاد أو عمل "اختبار التشغيل الذاتي" [19] POST ونقل التحكم إلى القرص الثابت. علما أن الأجهزة التي تحاكي القرص الثابت أثناء إقلاع النظام تتضمن أيضا سجل إقلاع لأنها تملك أيضا جدول أقسام. حتى و إن كانت لا تقبل الإقلاع.

نظام BIOS لن يقلع قرص MBR إذا لم يكن موجود ضمن معلومات ذاكرة CMOS. وينبغي أن يكون القرص بتهيئة صحيحة أيضا. من ناحية أخرى، حتى و إن لم يكن القرص ضمن سلسلة الإقلاع، ولكن كان يملك رقم للقرص، يمكن لأي برنامج يعمل في النمط الحقيقي [38] مثل سجل إقلاع أخر أو محمل إقلاع، تحميل وإقلاع سجل إقلاع ذلك القرص مباشرة. (راجع: إقلاع النظام).

دعم وسائط التخزين المقسمة، وبالتالي اعتماد MBR ظهر في مارس/آذار عام 1983، في نظام IBM PC-DOS 2.0 مع استخدام قرص ثابت بحجم 10 ميغابايت من شركة سيجيت في حاسوب IBM XT الجديد آنذاك والذي تضمن أيضا ذاكرة RAM بحجم ابتدائي 128 كيلوبايت [48] ومعالج إنتيل 8088، مع استخدام نظام ملفات FAT12.

النسخة الأصلية من سجل MBR كتبها ديفيد ليتون من شركة أي بي أم، في يونيو/حزيران عام 1982. جدول الأقسام [26][65] كان يدعم أربعة أقسام أولية، يمكن للنظام دوس آنذاك استخدام منها قسم واحد فقط. ولم تتغير هذه البنية في نظام التشغيل التالي DOS 3.0 مع نظام الملفات FAT16 . (أنظر أسفل: للطرح الست عشري/أسكي)

القسم الممتد، [32] وهو نوع خاص من الأقسام الأولية يعمل كحاوية للأقسام الأخرى، ظهر في DOS 3.2، والأقراص المنطقية [33] في القسم الممتد تم اعتمادها في DOS 3.30. أنظمة MS-DOS و PC DOS و OS/2 وويندوز لم تستخدم أبدا الأقسام المنطقية في الإقلاع، لذلك ظلت بنية سجل الإقلاع الرئيسي وشفرة الإقلاع وظيفيا من دون أي تغيير تقريبا، باستثناء بعض التطبيقات من الطرف الثالث، طوال فترة استخدام أنظمة دوس وأنظمة OS/2 حتى عام 1996. (أنظر أسفل: للطرح الأول والثاني).

في عام 1996 ظهرت عنونة الكتل المنطقية LBA[31] في أنظمة ويندوز 95B ودوس DOS 7.10 لدعم الأقراص الأكبر من 8 جيجابايت. وظهرت كذلك الأختام الزمني للقرص [26] (رغم أن الغرض الفعلي منها غير موثق [3] هذا أيضا يعكس فكرة أن سجل الإقلاع الرئيسي قصد منه في البداية أن يكون مستقل عن نظام التشغيل ونظام الملفات.

مع ذلك، قاعدة التصميم هذه تم تعديلها جزئيا في تطبيقات مايكروسوفت الأخيرة من سجل الإقلاع الرئيسي، التي فرضت استخدام طريقة النفاذ إلى القرص [31] CHS مع أقسام FAT16B و FAT32، (نوع 06h/0B h) بينما استخدمت LBA مع أقسام FAT16X و FAT32X (نوع 0Eh/0Ch).

رغم ضعف توثيق بعض تفاصيل MBR (التي أحيانا كانت سببا لمشاكل في التوافق)، تم اعتمد سجل الإقلاع الرئيسي على نطاق واسع نتيجة شعبية الأجهزة المتوافقة مع الحاسوب الشخصي وطبيعته الشبه ثابتة لسنوات عدة. لدرجة أنه تم دعمه في أنظمة تشغيل منصات أخرى. وأحيانا كان يستخدم إلى جانب المعايير الموجودة مسبقا أو متعددة المنصات في الإقلاع وتقسيم الوسيط.[4]

مدخلات الأقسام وشفرة إقلاع MBR المستخدمة في أنظمة التشغيل التجارية، مقيدة بـ 32 بت [2]، لهذا السبب الحجم الأقصى للقرص المعتمد باستخدام 512 بايت في القطاع هو في حدود 2 تيرابايت [2] (سواء كان ذلك في المحاكاة أو فعليا) مع مخطط تقسيم القرص MBR (دون استخدام الطرق الغير معيارية). لذلك، ظهرت الحاجة إلى مخطط آخر في تقسيم الأقراص الكبيرة التي ظهرت في السوق عام 2010. هذه المخطط الجديد يدعى GPT؛ وهو خليفة MBR، ورغم أنه لا يوفر توافق للإصدارات السابقة مع الأنظمة التي لا تدعمه. لكنه يوظف نسخة للحماية هجينة من MBR، تدعى اصطلاحا protective MBR، الهدف منها ضمان عملية التكامل (سلامة البيانات). (راجع: كتاب "جدول أقسام GUID")

صيغ أخرى من سجل الإقلاع الرئيسي الهجين hybrid MBR تم تصميمها وتنفيذها من قبل أطراف أخرى من أجل الحفاظ على الأقسام الواقعة في منطقة 2 تيرابايت الأولى في القرص في كلا مخططي التقسيم "بالتوازي" و/أو من أجل السماح لأنظمة التشغيل القديمة الإقلاع من أقسام GPT. لكن هذه الصيغ الغير معيارية يمكنها أيضا أن تسبب مشاكل في التوافق.

الطرح التالي يعرض لأول MBR (ضمن في FDISK) نظام IBM PC DOS 2.00 من عام 1983، وفي الإصدارات 2.10، 3.00، 3.10، 3.20، 3.21 حتى DOS 3.30 التي تغيرت فيها الشفرة قليلا.

القطاع المطلق 0 (الفيزيائي) (CHS 0-0-1)

	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	0123456789ABCDEF
0000	FA	33	C0	8E	D0	BC	00	7C	8B	F4	50	07	50	1F	FB	FC	[.3.....|..P.P...]
0010	BF	00	06	B9	00	01	F3	A5	EA	1D	06	00	00	BE	BE	07	[................]
0020	B3	04	80	3C	80	74	0E	82	3C	00	75	1C	83	C6	10	FE	[...<.t..<.u.....]
0030	CB	75	EF	CD	18	8B	14	8B	4C	02	8B	EE	83	C6	10	FE	[.u......L.......]
0040	CB	74	1B	82	3C	00	74	F4	BE	8B	06	32	ED	AC	8A	C8	[.t..<.t....2....]
0050	AC	56	BB	07	00	B4	0E	CD	10	5E	E2	F4	EB	FE	BF	05	[.V......^.......]
0060	00	BB	00	7C	B8	01	02	57	CD	13	5F	73	0C	33	C0	CD	[..|...W._s.3...]
0070	13	4F	75	ED	BE	A3	06	EB	D2	BE	C2	06	81	3E	FE	7D	[.Ou.........>.}]
0080	55	AA	75	C7	8B	F5	EA	00	7C	00	00	17	49	6E	76	61	[U.u.....|...Inva]
0090	6C	69	64	20	70	61	72	74	69	74	69	6F	6E	20	74	61	[lid partition ta]
00A0	62	6C	65	1E	45	72	72	6F	72	20	6C	6F	61	64	69	6E	[ble.Error loadin]
00B0	67	20	6F	70	65	72	61	74	69	6E	67	20	73	79	73	74	[g operating syst]
00C0	65	6D	18	4D	69	73	73	69	6E	67	20	6F	70	65	72	61	[em.Missing opera]
00D0	74	69	6E	67	20	73	79	73	74	65	6D	41	75	74	68	6F	[ting systemAutho]
00E0	72	20	2D	20	44	61	76	69	64	20	4C	69	74	74	6F	6E	[r - David Litton]
00F0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[................]
0100	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[................]
0110	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[................]
0120	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[................]
0130	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[................]
0140	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[................]
0150	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[................]
0160	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[................]
0170	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[................]
0180	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[................]
0190	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[................]
01A0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[................]
01B0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	80	01	[................]
01C0	01	00	0B	7F	BF	FD	3F	00	00	00	C1	40	5E	00	00	00	[......?....@^...]
01D0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[................]
01E0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[................]
01F0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	55	AA	[..............U.]
	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	0123456789ABCDEF

- شفرة تنفيذية [44]، في أول 139 بايت (من 00h إلى 8Ah)
- رسائل أخطاء 80 بايت (من 8Bh إلى DAh)
- توقيع الكاتب / عبارة "Author - David Litton" التي حذفت في MBR DOS 3.30
- 206 بايت حشو بايت صفر [43] (من 0F0h إلى 1BDh) عن طريق FDISK
- جدول أقسام 64-بايت، (من 1BEh إلى 1FDh)؛ منطقة البيانات تعتمد على نظام الملفات وحجم، وبنية القرص
- توقيع إقلاع، 2-بايت AA55h أو الرقم السحري في نهاية القطاع؛ (في أنظمة إنتل، الكلمات الست عشرية (16-بت) تخزن بحيث البايت المنخفض أولا والأعلى أخيرا) [58]

الشفرة المعيارية MBR! المستخدمة منذ MS-DOS 3.30 وحتى ويندوز 95A قبل 95B، (مضمنة في FDISK)، (باستثناء جزء صغير، هذه تطابق شفرة 2.00 IBM PC DOS من عام 1983).

القطاع المطلق 0 (الفيزيائي) (CHS 0-0-1)

	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	0123456789ABCDEF
0000	FA	33	C0	8E	D0	BC	00	7C	8B	F4	50	07	50	1F	FB	FC	[.3.....|..P.P...]
0010	BF	00	06	B9	00	01	F2	A5	EA	1D	06	00	00	BE	BE	07	[................]
0020	B3	04	80	3C	80	74	0E	80	3C	00	75	1C	83	C6	10	FE	[...<.t..<.u.....]
0030	CB	75	EF	CD	18	8B	14	8B	4C	02	8B	EE	83	C6	10	FE	[.u......L.......]
0040	CB	74	1A	80	3C	00	74	F4	BE	8B	06	AC	3C	00	74	0B	[.t..<.t.....<.t.]
0050	56	BB	07	00	B4	0E	CD	10	5E	EB	F0	EB	FE	BF	05	00	[V.......^.......]
0060	BB	00	7C	B8	01	02	57	CD	13	5F	73	0C	33	C0	CD	13	[..|...W._s.3...]
0070	4F	75	ED	BE	A3	06	EB	D3	BE	C2	06	BF	FE	7D	81	3D	[Ou..........}.=]
0080	55	AA	75	C7	8B	F5	EA	00	7C	00	00	49	6E	76	61	6C	[U.u.....|..Inval]
0090	69	64	20	70	61	72	74	69	74	69	6F	6E	20	74	61	62	[id partition tab]
00A0	6C	65	00	45	72	72	6F	72	20	6C	6F	61	64	69	6E	67	[le.Error loading]
00B0	20	6F	70	65	72	61	74	69	6E	67	20	73	79	73	74	65	[operating syste]
00C0	6D	00	4D	69	73	73	69	6E	67	20	6F	70	65	72	61	74	[m.Missing operat]
00D0	69	6E	67	20	73	79	73	74	65	6D	00	00	00	00	00	00	[ing system......]
00E0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[................]
00F0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[................]
0100	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[................]
0110	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[................]
0120	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[................]
0130	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[................]
0140	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[................]
0150	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[................]
0160	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[................]
0170	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[................]
0180	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[................]
0190	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[................]
01A0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[................]
01B0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	80	01	[................]
01C0	01	00	0B	7F	BF	FD	3F	00	00	00	C1	40	5E	00	00	00	[......?....@^...]
01D0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[................]
01E0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[................]
01F0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	55	AA	[..............U.]
	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	0123456789ABCDEF

- شفرة تنفيذية [44]، أول 139 بايت (من 00h إلى 8Ah)
- رسائل أخطاء، في 80 بايت (من 8Bh إلى DAh) [60]
- 227 بايت حشو [43] بايت الصفر (من DBh إلى 1BDh) عن طريق FDISK.
- عادة لا تستخدم، (باستثناء بعض الأنظمة مثل NetBSD) [42]
- جدول أقسام 64-بايت (من 1BEh إلى 1FDh) هذه المنطقة تعتمد على نظام الملفات وحجم وبنية القرص
- توقع إقلاع، 2-بايت في نهاية القطاع أيضا يدعى الرقم السحري AA55h؛ في أنظمة إنتل، الكلمات الست عشرية (16-بت) تخزن بحيث يكون البايت المنخفض أولا والأعلى أخيرا [58]

الإضافات التالية ظهرت في الأنظمة اللاحقة:

- أنظمة 95B/98/98SE/ME. تستخدم الختم الزمني 6 بايت (من 0DAh إلى 0DFh)، أي إذا استخدمت هذه الشفرة في إحدى تلك الأنظمة، ستتغير 4 بايت (التي من 0DCh إلى 0DFh) في الإقلاع التالي [29]
- لغة الإصدارة ؛ ظهرت فقط في أنظمة ويندوز الأحدث [37]
- توقيع قرص، 4 بايت (من 1B8h إلى 1BBh)، رقم تسلسلي في ويندوز أن تي [36] (وجود هذا التوقيع دليل على وجود أحد أنظمة ويندوز أن تي، XP/2000.)

DAh	DBh	DCh	DDh	DEh	DFh	حيود الختم الزمني ←
00	00	81	08	05	07	مثال : 4 بايت تعكس رقم القرص وزمن كتابتها إلى سجل الإقلاع :
دائما أصفار		رقم القرص	ثواني	دقائق	ساعات	(بترتيب معكوس)

هذه الشفرة المعيارية! (قبل ويندوز 95B) كان يكتبها برنامج دوس FDISK /MBR إلى القطاع الأول على القرص الثابت. هذه الشفرة تطابق شفرة محمل الإقلاع المعياري في مدير أقسام رانيش RPM وفي العديد من أدوات MBR الأخرى [44]. لكن منذ صدور نظام ملفات FAT32 (ويندوز 95B) أصبحت شفرة FDISK أكثر تعقيد من هذه.

هذه الشفرة أيضا مستقلة عن نظام تشغيل، أي يمكن استخدامها في إقلاع أي نظام يرتكز على معالج x86. طالما توفرت فيه الشروط التالية:

1. أن يكون نظام التشغيل على القرص الثابت الأول الرئيسي.
2. أن يكون هناك قسم نشيط واحد فقط.
3. أن يكون قطاع إقلاع القسم عند أو تحت الأسطوانة 1024 على القرص (لأن روتين INT 13 مقيد بتلك القيمة عند قراءة القرص).
4. أن يكون محمل إقلاع النظام في القطاع الأول للقسم.

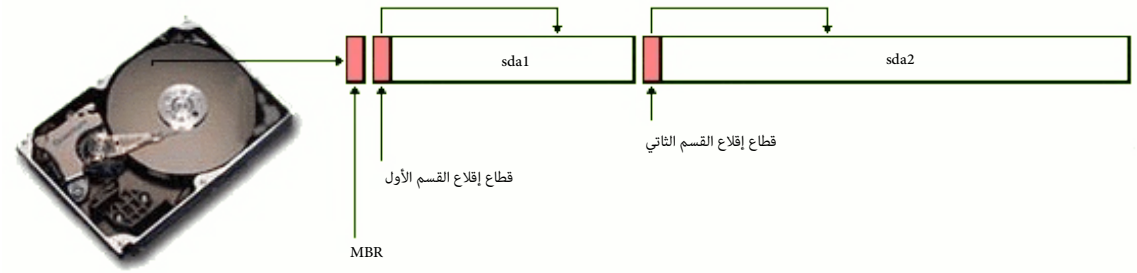
بعد فحص العتاد عن طريق برنامج التشغيل الذاتي POST [19]، شفرة BIOS [20] تحمل قطاع MBR عند عنوان الذاكرة 0000:**7C00** ثم تنفذه بقفزة JMP إلى الشفرة المنسوخة هناك، لكن بخلاف قطاع إقلاع VBR، هذه الشفرة يجب أولا أن تنقل نفسها إلى عنوان 0000:**0600**. هذا ضروري لأن، فيما بعد [21] MBR سيحمل قطاع إقلاع القسم النشيط في نفس المنطقة.

تقسيم القرص

في بداية ثمانيات القرن الماضي، قدمت شركة IBM مع نظام PC DOS 2.0، أداة تدعى FIDSK تستخدم في إنشاء وصيانة أقسام MBR. ووفقا لذلك المخطط، عند تقسيم جهاز التخزين، MBR سوف يتضمن على جدول أقسام MPT يصف مواقع، وأحجام، وخصائص المناطق الخطية الأخرى التي يشار إليها بالأقسام (أو وحدات تخزين منطقية) [33]، (أنظر للشكل أدناه).

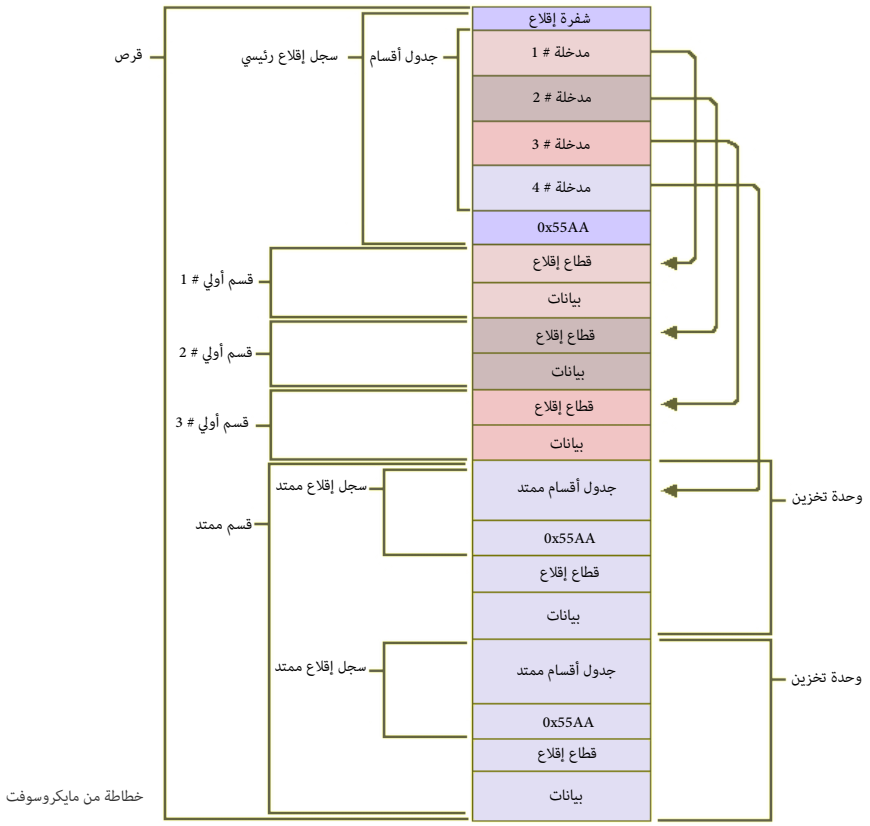
الأقسام يتعامل معها كأنها أقراص حقيقية متعددة، ويمكن تهيئة كل قسم منها بنظام ملفات مختلف، أو استخدامها لأي غرض آخر. لكن حتى يعمل نظام التشغيل يجب تقسيم القرص إلى قسم واحد على الأقل مع تهيئته. القسم يتألف من سلسلة أسطوانات على القرص الثابت، كل قسم محدد ببداية ونهاية أسطوانة معينة [47] (لكن حجم الأسطوانات يتفاوت من قرص لأخر). البرنامج المسؤول عن إنشاء وحذف وتحجيم الأقسام ومعالجتها على القرص يدعى غالبا محرر أقسام .

الأقسام نفسها أيضا يمكنها أن تتضمن على بيانات تصف مخططات تقسيم للقرص أكثر تعقيد، مثال على ذلك، سجلات إقلاع القسم الممتد EBR، [34] أو سجلات أقسام أنظمة بيركلي BSD disklabels، [42] أو أقسام ميتاداتا (البيانات الوصفية) في مدير الأقراص المنطقية LDM. [9] [17]



كما ذكرنا، سجل الإقلاع الرئيسي يقع دائما في القطاع الأول من القرص (عند الحيد الفيزيائي 0)، ولا يمكن أن يقع أبدا على أي قسم. ولذلك هو يسبق القسم الأول على القرص. علما أن قطاع الإقلاع في الأجهزة التي بدون تقسيم أو في القسم المنفرد يسمى: سجل إقلاع القسم VBR.

في الأجهزة التي توظف التقنية البرمجية DDO BIOS overlay [57] أو برامج إدارة الإقلاع، يمكن تحريك جدول الأقسام إلى موقع فيزيائي أخر على القرص؛ مثال على ذلك، Ontrack Disk Manager الذي غالبا ما يضع نسخة أصلية من MBR في القطاع الثاني من القرص، ثم يخفي نفسه عن أي إقلاع للنظام أو تطبيق. ويتم التعامل مع نسخة MBR كما لو أنها تقع في القطاع الأول.



4 مدخلات أولية في جدول أقسام MBR، إحداها مدخلة قسم ممتد [32]، قطاعات الإقلاع في الأقسام والأقراص المنطقية [33] (وحدات التخزين).تملك نفس تعريف الحقول في جدول الأقسام MBR و EBR[34].

تخطيط القطاع

رسميا، هناك 4 مدخلات أولية للأقسام [26] في مخطط جدول أقسام MBR، رغم أن بعض أنظمة التشغيل وأدوات النظام رفعت عددها إلى 5 مدخلات، كما في أقسام AAPs [30] في DR-DOS 7.07 [10]_PTS-DOS 6.60، أو 8 مدخلات في SpeedStor و AST و 3.x NEC MS-DOS [11] [12]، أو حتى 16 مدخلة، في مدير الأقراص Ontrack أنظمة MS-DOS و PC DOS.

بنية سجل الإقلاع الرئيسي (التقليدية)				عنوان	
وصف		حجم (بايت)		عشري	ست عشري
منطقة شفرة الإقلاع		436 (الأقصى 446)		+0	+000h
توقيع قرص (اختياري) [5] [6] [7] [8] [36]		4 (حتى 10)		+440	+1B8h
بلا قيم (عادة)	0x000	2		+444	+1BCh
جدول أقسام أولية	مدخلة قسم #1	16		+446	+1BEh
	مدخلة قسم #2	16		+462	+1CEh
	مدخلة قسم #3	16		+478	+1DEh
	مدخلة قسم #4	16		+494	+1EEh
توقيع إقلاع [1]	55h	2		+510	+1FEh
	AAh			+511	+1FFh
حجم إجمالي		512			

بنية سجل الإقلاع الرئيسي المعيارية (الحديثة).			عنوان	
وصف		حجم (بايت)	عشري	ست عشري
منطقة شفرة الإقلاع (جزء 1)		218	+0	+000h
ختم زمني للقرص [3] [29] [2] (اختياري، في MS-DOS 7.1–8.0 و ويندوز 95B/98/98SE/ME. كبديل، يمكن أن يخدم كتوقيع OEM loader مع NEWLDR)	0000h	2	+218	+0DAh
	القرص الفيزيائي الأصلي (80h-FFh)	1	+220	+0DCh
	ثواني (0-59)	1	+221	+0DDh
	دقائق (0-59)	1	+222	+0DEh
	ساعات (0-23)	1	+223	+0DFh
منطقة شفرة الإقلاع (جزء 2، مدخلة الشفرة عند +000h)		216 (أو 222)	+224	+0E0h
توقيع قرص [5] [36] (اختياري، أنظمة UEFI و ويندوز NT/2000/Vista/7 والأنظمة الأخرى)	توقيع قرص 32-بت	4	+440	+1B8h
	0000h (إذا كان محمي من النسخ = 5A5Ah)	2	+444	+1BCh
جدول أقسام أولية	مدخلة قسم #1	16	+446	+1BEh
	مدخلة قسم #2	16	+462	+1CEh
	مدخلة قسم #3	16	+478	+1DEh
	مدخلة قسم #4	16	+494	+1EEh
توقيع إقلاع [1]	55h	2	+510	+1FEh
	AAh		+511	+1FFh
حجم إجمالي		512		

بنية سجل الإقلاع الرئيسي AAP MBR - عائلة نظام دوس [30]			عنوان	
وصف		حجم (بايت)	عشري	ست عشري
منطقة شفرة الإقلاع		428	+0	+000h
توقيع AAP (اختياري)	78h	2	+428	+1ACh
	56h		+429	+1ADh
سجل AAP (اختياري) (مدخلة قسم #0 في AAP لها دلالات خاصة)	القرص الفيزيائي AAP (80h-FEh)، (غير مستعملة: 00h). (محجوزة: 01h-7Fh, FFh)	1	+430	+1AEh
	CHS (بداية) عنوان ملف صورة/قسم AAP أو VBR/EBR	3	+431	+1AFh
	محجوزة من أجل نوع القسم AAP (غير مستعملة: 00h) (اختيارية)	1	+434	+1B2h
	محجوزة من أجل عنوان نهاية CHS في AAP (اختيارية؛ البايت عند الحيد +1B5h يستعمل أيضا في تدقيق مجموع MBR في برنامج (PTS DE, BootWizard)؛ غير مستعملة تأخذ: 000000h)	3	+435	+1B3h
	بداية LBA لملف صورة AAP أو VBR/EBR أو القطاعات النسبية للقسم AAP (المنسوخة إلى الحيد +01Ch في القطاع المحمل فوق مدخلة "القطاعات المخفية" في كتلة	4	+438	+1B6h

	DOS 3.31 BPB (أو ما يحاكي ذلك) لدعم أيضا إقلاع EBR)			
محجوزة للقطاعات في AAP (اختيارية؛ غير مستعملة تأخذ: 00000000h)	4	+442	+1BAh	
جدول أقسام أولية	مدخلة قسم #1	16	+446	+1BEh
	مدخلة قسم #2	16	+462	+1CEh
	مدخلة قسم #3	16	+478	+1DEh
	مدخلة قسم #4	16	+494	+1EEh
توقيع إقلاع [1]	55h	2	+510	+1FEh
	AAh		+511	+1FFh
حجم إجمالي		512		

بنية سجل الإقلاع الرئيسي NEWLDR MBR			عنوان	
	وصف	حجم (بايت)	عشري	ست عشري
سجل NEWLDR (اختياري)	حجم سجل NEWLDR / تعليمة قفزة JMPS (EBh) (غالبا، تأخذ القيم 0Ah/16h/1Ch من أجل الشفرة التي تبدأ عند الحيود +00Ch/+018h/+01Eh)	2	+0	+000h
	توقيع محمل الإقلاع "NEWLDR"	6	+2	+002h
	القرص الفيزيائي وعلم الإقلاع الخاص بوسيلة الإقلاع LOADER (80h-FEh, 00h-7Eh, FFh, 7Fh) (إذا لم تستعمل، هذه و 3 بايتات التالية جميعا يجب أن تكون 0)	1	+8	+008h
	عنوان CHS الخاص بقطاع إقلاع وسيلة LOADER أو ملف الصورة (مثال، شفرة محمل الإقلاع IBMBIO.LDR) (غير مستعملة تأخذ: 000000h)	3	+9	+009h
	القيمة الأدنى المسموح بها للتسجيل DL ، ما عدا ذلك تأخذ من جدول الأقسام (القيمة الاعتيادية: 80h)، (استخدام دائما DL : 00h)، (استخدام دائما مدخلة الجدول: FFh)	1	+12	+00Ch
	محجوزة (القيمة الاعتيادية: 000000h)	3	+13	+00Dh
	LBA الخاص بقطاع إقلاع وسيلة LOADER أو ملف الصورة (اختيارية، غير مستعملة تأخذ: 00000000h)	4	+16	+010h
	حيد رقعة (Patch offset) خاص بوحدة إقلاع VBR (القيمة الاعتيادية: 0000h إذا كانت غير مستعملة)، (ما عدا ذلك تأخذ 0024h أو 01FDh)	2	+20	+014h
	تدقيق المجموع (غير مستعملة تأخذ القيمة 0000h)	2	+22	+016h
	توقيع محمل صانع المعدات الأصلية OEM loader ("MSWIN4" من أجل نظام تشغيل REAL/32، أنظر أيضا الحيد +0DAh، المقترن بلصيقة صانع المعدات الأصلية OEM label عند الحيد +003h في VBRs) (اختيارية)	6	+24	+018h
منطقة شفرة الإقلاع (مدخلة الشفرة عند +000h)		متفاوت	متفاوت	متفاوت
توقيع AAP (اختياري)	78h	2	+428	+1ACh
	56h		+429	+1ADh
سجل AAP (اختياري)	مدخلة القسم #0 في AAP، لها دلالات خاصة.	16	+430	+1AEh
جدول أقسام أولية	مدخلة قسم #1	16	+446	+1BEh
	مدخلة قسم #2	16	+462	+1CEh
	مدخلة قسم #3	16	+478	+1DEh
	مدخلة قسم #4	16	+494	+1EEh
توقيع إقلاع [1]	55h	2	+510	+1FEh
	AAh		+511	+1FFh
حجم إجمالي		512		

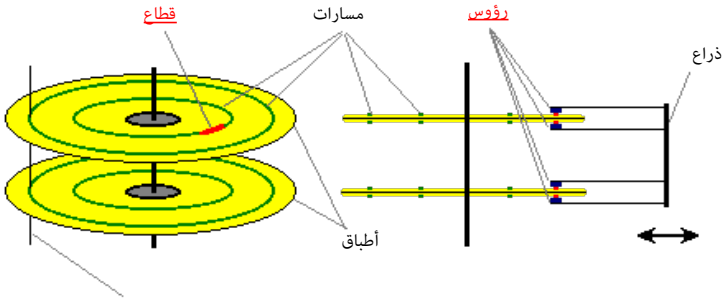
بنية سجل الإقلاع الرئيسي في AST/NEC MS-DOS و SpeedStor			عنوان	
	وصف	حجم (بايت)	عشري	ست عشري
منطقة شفرة الإقلاع		380	+0	+000h
توقيع AST/NEC (اختياري، وليس من أجل وسيلة SpeedStor)	5Ah	2	+380	+17Ch
	A5h		+381	+17Dh
جدول أقسام ممتد في AST/NEC (اختياري، أيضا من أجل وسيلة SpeedStor)	مدخلة قسم #8	16	+382	+17Eh
	مدخلة قسم #7	16	+398	+18Eh
	مدخلة قسم #6	16	+414	+19Eh

	مدخلة قسم #5	16	+430	+1AEh
جدول أقسام أولية	مدخلة قسم #4	16	+446	+1BEh
	مدخلة قسم #3	16	+462	+1CEh
	مدخلة قسم #2	16	+478	+1DEh
	مدخلة قسم #1	16	+494	+1EEh
توقيع إقلاع [1]	55h	2	+510	+1FEh
	AAh		+511	+1FFh
حجم إجمالي		512		

بنية سجل مدير القرص DM MBR			عنوان	
وصف		حجم (بايت)	عشري	ست عشري
منطقة شفرة الإقلاع		252	+0	+000h
توقيع مدير القرص DM (اختياري)	AAh	2	+252	+0FCh
	55h		+253	+0FDh
جدول أقسام ممتد في مدير القرص DM (اختياري)	مدخلة قسم	16	+254	+0FEh
	مدخلة قسم	16	+270	+10Eh
	مدخلة قسم	16	+286	+11Eh
	مدخلة قسم	16	+302	+12Eh
	مدخلة قسم	16	+318	+13Eh
	مدخلة قسم	16	+334	+14Eh
	مدخلة قسم	16	+350	+15Eh
	مدخلة قسم	16	+366	+16Eh
	مدخلة قسم	16	+382	+17Eh
	مدخلة قسم	16	+398	+18Eh
	مدخلة قسم	16	+414	+19Eh
	مدخلة قسم	16	+430	+1AEh
جدول أقسام أولية	مدخلة قسم #1	16	+446	+1BEh
	مدخلة قسم #2	16	+462	+1CEh
	مدخلة قسم #3	16	+478	+1DEh
	مدخلة قسم #4	16	+494	+1EEh
توقيع إقلاع [1]	55h	2	+510	+1FEh
	AAh		+511	+1FFh
حجم إجمالي		512		

مدخلات جدول الأقسام

مدخلات [26] جدول الأقسام MPT هي تقنية في القرص الثابت من زمن حاسوب آي بي إم PC XT. حين كان يقسم وسيط التحزين باستعمال وحدات من أسطوانات ورؤوس وقطاعات، هذه الثوابت الثلاثة كانت طريقة للعنونة كتل البيانات على القرص الثابت، عرفت اختصارا [31] بمتتابعة CHS. اليوم استخدام هذه القيم غير مرتبط فيزيائيا بالأقراص الحديثة الكبيرة. بل ليست موجودة أصلا في أقراص الحالة الصلبة الحديثة SSD لأن هذه الأخيرة لا تملك أصلا أسطوانات أو رؤوس فيزيائية (مادية).



شكل 7

أسطوانة : مجموعة من المسارات التي يمكن النفاذ إليها بدون تحريك الذراع

كما هو متفق عليه، في مخطط [31] CHS، فهرسة القطاع تبدأ من القطاع 1 وليس 0. ونتيجة لوجود خطأ في كافة إصدارات MS-DOS/PC DOS حتى الإصدار 7.10، عدد الرؤوس أصبح مقيد برقم 255 بدل 256.(وفقا لبعض المصادر [7])، واليوم عندما يكون عنوان CHS كبير ولا يتناسب مع هذه الحقول تستخدم المتتابعة (1023 254 63) (راجع أسفل تحليل:**FE FF FF**).

في الأنظمة وأدوات القرص القديمة، قيمة الأسطوانة غالبا ما تلتف حول[22] تردد (مودولو) [23] حاجز CHS القريب من 8 ضحايات [46] (أحد قيود BIOS)، الذي يسبب التباس للمستخدمين وربما تلف في البيانات. (في حالة سجل protective MBR على قرص GPT، مواصفة إنتل EFI تتطلب استخدام المتتابعة 1023, 255, 63) راجع أيضا كتيب "جدول أقسام GUID".

قيمة أسطوانة 10-بت مسجلة ضمن 2 بايت لتسهيل عمل النداءات إلى روتينات النفاذ للقرص BIOS INT 13h، بحيث 16 بت مجزئة على قطاعات وأسطوانات، وليست على حدود البايت [14]

ونتيجة لمحدودية عنونة CHS [18][17][31] تم التحول إلى نظام عنونة الكتل المنطقية LBA. الذي فيه طول القسم وعنوان بداية القسم كلاهما قيم للقطاع مخزنة في مدخلات جدول الأقسام باستخدام قيم 32 بت (أنظر للمثال). حجم القطاع 512 (2^9) بايت دائما ثابت، وهذه القيمة جزء من الشفرة الداخلية [24] في العديد من مكونات الحاسب، التي تشمل مجموعات الشرائح، قطاعات الإقلاع، أنظمة التشغيل، محركات قواعد البيانات، أدوات تقسيم القرص، أدوات نظام الملفات، والنسخ الاحتياطي، وبرمجيات أخري.

منذ نهاية 2009، أصبح متوفرا في الأسواق محركات أقراص (بتقنية جديدة تدعى AF أو 4K)، توظف قطاع بحجم 4,096 بايت أو 4 كيلو بايت، رغم أن بعض هذه الأقراص لا تزال تعلن في النظام المضيف عن حجم القطاع 512 بايت، (نتيجة عملية تحويل يقوم بها البرنامج الثابت في القرص)، من أجل تجنب حالات عدم التوافق في التطبيقات، هذه النوع من الأقراص يشار له باسم أقراص محاكاة قطاع 512 (لمعلومات أكثر راجع موضوع 512e في الموسوعة الحرة – النسخة الانجليزية).

بما أن عناوين وأحجام الكتل تخزن في جدول أقسام MBR باستخدام 32 بت، فإن الحجم الأقصى وكذلك أعلى عنوان بداية للقسم في أقراص قطاع 512 بايت (سواء في المحاكاة أو فعليا) لا يمكن أن يتجاوز 2 تيرابايت – 512 بايت (2,199,023,255,040 بايت أو 4,294,967,295 (2^{32} - 1) قطاع × 512 (2^9) بايت لكل قطاع).[2] لذلك كان تجاوز مشكلة حد 2 تيرابايت من الدوافع الرئيسية لتطوير جدول أقسام معرفات GUID أو GPT.

بما أن معلومات التقسيم تخزن في جدول أقسام MBR باستخدام طول وعنوان كتلة البداية، قد يكون ممكن ولو نظريا تحديد الأقسام بالطريقة تكون فيها المساحة المخصصة للقرص الذي يستخدم قطاع 512 بايت، تعطي حجم كلي يقارب 4 تيرابايت، إذا كانت جميع الأقسام باستثناء قسم واحد تقع تحت حد 2 تيرابايت وتم تعيين القسم الأخير ليبدأ عند أو قرب الكتلة 2^{32} - 1 مع تخصيص حجم مثلا حتى 2^{32} - 1 ، وبهذا، يمكن تعريف قسم سوف يحتاج في النفاذ إلى عنوان القطاع، 33 بت بدل 32 بت. لكن، عمليا هذا تدعمه فقط أنظمة التشغيل معينة التي تمكن LBA-48، وتستخدم داخليا عناوين القطاع 64-بت، هذه الأنظمة تشمل لينكس، في بي أس دي، وبندوز 7 [19].

نظرا لضيق المساحة المخصصة للشفرة [44] وطبيعة جدول أقسام MBR الذي يدعم فقط 32 بت، قطاعات الإقلاع، حتى وإن كانت تدعم الحجم LBA-48 بدلا من LBA-28، غالبا، سوف تستخدم حسابات 32-بت، إلا إذا كانت مصممة لدعم كامل نطاق عنونة LBA-48 أو قصد منها العمل فقط على منصات 64-بت. و أي شفرة إقلاع أو نظام تشغيل يستخدم داخليا عناوين القطاع 32-بت حتما سيجعل العناوين تلتف عند النفاذ إلى هذا القسم، وبالتالي سيؤدي ذلك إلى تلف البيانات على كافة الأقسام.

هناك قيود كذلك على الأقراص التي تعلن عن حجم قطاع مخالف للحجم 512 بايت، مثل الأقراص الخارجية (التي تستخدم الناقل التسلسلي العام USB). قطاع بحجم 4,096 ينتج عنه زيادة ثمانية أضعاف في حجم القسم الذي يمكن تعريفه باستخدام MBR ، هذا سيسمح بوصول حجم الأقسام إلى 16 تيرابايت (2^{32} × 4096 بايت) [20] إصدارات ويندوز الأحدث من ويندوز XP تدعم كذلك أحجام قطاع أكبر مثلها مثل نظام ماك عشرة Mac OS X، ونواة لينكس التي تدعم أحجام القطاع الأكبر منذ إصدارة 2.6.31 [21] أو إصدارة 2.6.32، [22]، لكن مشاكل مع محملات الإقلاع وأدوات تقسيم القرص وتطبيقات نظام BIOS أوجدت بعض القيود، [23] غالبا لأنها [24] مبرمجة لحجز فقط 512 بايت من أجل **صوانات** القطاع، وهذا يتسبب في إعادة كتابة الذاكرة من أجل أحجام القطاع الأكبر. وقد يسبب هذا سلوك للنظام غير متوقع، ينبغي تجنبه إذا كانت مسألة التوافق والالتزام بالمعيار مطلوبة.

عند تقسيم جهاز تخزين البيانات باستخدام GPT، سجل الحماية protective MBR سيظل يحتوي على جدول أقسام MBR، لكن الغرض الوحيد من وجوده سيكون الإشارة إلى وجود GPT ومنع برامج إدارة القرص التي تفهم فقط جدول أقسام MBR التقليدي من إنشاء أي أقسام في المساحة التي تظن أنها فارغة على القرص، (أي منعها من محو بالخطأ مخطط جدول أقسام GPT).

بنية إحدى مدخلات جدول أقسام MBR- مدخلة 16-بايت [13] (الحقول متعددة البايت ستكون بترتيب نهوي صغير)

وصف	قيم		طول الحقل	إزاحة (نظام ست عشري)	
	مثال	معنى	(بايت)	نسبة (في المدخلة)	في القطاع
حقل الحالة أو القرص الفيزيائي (تعيين بت 7 من أجل القرص النشط / يقبل الاقلاع) سجل MBR القديم يقبل فقط القيم التالية قرص نشيط: 80h قرص غير نشيط: 00h القيم من 01h إلى 7Fh غير صالحة (أو باطلة) [3]	80	قابل للإقلاع	1	+0h	+0Eh
عنوان CHS للقطاع المطلق الأول في القسم. [4]. الصيغة ممثلة في ثلاثة بايتات، تظهر في الصفوف الثلاثة التالية.	01 01 00	0, 1, 1	3	1h - 3h	0Fh- 01h
رأس / جانب [5] h 7-- 0 x x x x x x x x	01		1	+1h	+0Fh
القطاع في بتات 0 -5 بتات 6 -7 هي بتات عليا في الأسطوانة [5] c 8-9 s 5- 0 x x x x x x x x	01		1	+2h	+00h
بتات 0 –7 في الأسطوانة [5] c 7– 0 x x x x x x x x	00		1	+3h	+01h
نوع القسم [15]	DE	Dell Utility	1	+4h	+02h
عنوان CHS للقطاع المطلق الأخير في القسم. [4] الصيغة ممثلة في ثلاثة بايتات، تظهر في الصفوف الثلاثة التالية.	FE 3F 04	4، 254، 63	3	5h - 7h	03h -05h
رأس / جانب [5] h 7- 0 x x x x x x x x	FE		1	+5h	+03h
القطاع في بتات 0 -5 بتات 6 --7 هي بتات عليا في الأسطوانة [5] c 8-9 s 0-5 x x x x x x x x	3F		1	+6h	+04h
بتات 0 -7 في الأسطوانة c 0-7 x x x x x x x x	04		1	+7h	+05h
عنوان LBA للقطاع المطلق الأول في القسم. [6]	3F 00 00 00	63	4	8h - Bh	06h -09h
عدد القطاعات في القسم. [6]	86 39 01 00	80,262	4	Ch - Fh	0Ah- 0Dh

جدول 7

هذه البيانات في مدخلات جدول الأقسام MPT تخبر النظام عن مكان بداية ونهاية كل قسم على القرص، [47] وعن حجم القسم [63]، وما إذا كان القسم يقبل الإقلاع أم لا ، وأي نوع نظام ملفات يستخدم القسم. قيم بداية القطاع والرأس والأسطوانة تستخدمها فقط الأنظمة التي تشتغل في نمط عنونة CHS، [31] المعيارية في جميع الأقراص التي بحجم 8.4 جيجابايت أو أقل. لكن قيم CHS لا تعمل في أقراص أكبر من 8.4 جيجابايت ولا يمكنها تمثيل الأقسام على تلك الأقراص (راجع الأمثلة أسفل). يمكن عنونة الأقراص الأكبر فقط باستخدام نظام الكتل المنطقية LBA. [31] في هذه الحالة، بتم تجاهل قيم بداية القطاع والرأس والأسطوانة، وتستخدم فقط القيم في حقول مجموع القطاعات والقطاع النسبي. (تظهر في نهاية الجدول السابق).

حقل القطاع النسبي يشير بالضبط إلى عنوان LBA (القطاع المطلق) حيث يبدأ القسم [47]، بينما حقل مجموع القطاعات يشير إلى الطول، الذي دائما متواصل. وهكذا، من هاتان القيمتان يمكن للنظام معرفة بالضبط أين يقع القسم فيزيائيا على القرص.

جدول الأقسام المعياري [26] في MBR (يظهر أدناه) عبارة عن بنية بيانات بقيمة 64 بايت للتعريف بنوع وموقع الأقسام الأولية والقسم الممتد (إن وجد) [32] على القرص الثابت، وفقا لتخطيط معياري مستقل عن نظام التشغيل. كل مدخلة جدول أقسام بطول 16 بايت، العدد الأقصى في السجل التقليدي هو 4 مدخلات, كل مدخلة تبدأ عند حيد محدد سلفا في ومن بداية القطاع:

مدخلات	طول (بايت)	عنوان عشري	عنوان ست عشري
قسم #1	16	446 - 461	1BE - 1CD
قسم #2	16	462 - 477	1CE - 1DD
قسم #3	16	478 - 493	1DE - 1ED
قسم #4	16	494 - 509	1EE - 1FD

حجم وإزاحة جدول أقسام سجل الإقلاع الرئيسي

في هذا المثال التالي يظهر جزء من شفرة جدول أقسام MBR؛ يتضمن قسمين فقط. لحظ عندما يكون عدد الأقسام أقل من أربعة، بقية الحقول غالبا ما تكون محشوة بقيم الصفر[43]

```
01B0                                           00 01 [................]
01C0 01 00 DE FE 3F 04 3F 00 00 00 86 39 01 00 80 00 [..bb?.?...|9..|.]
01D0 01 05 07 FE FF FF C5 39 01 00 F8 AF 4E 09 00 00 [...byyA9..0 N...]
01E0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [................]
01F0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 55 AA [..............U.]
     00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F  0123456789ABCDEF
```

- علم إقلاع
- حقول قطاع بداية القسم (أسطوانة، رأس، قطاع)
- نوع القسم (مثل: 07h و DEh)
- حقول قطاع نهاية القسم (أسطوانة، رأس، قطاع)
- حقل القطاعات النسبية
- حقل قطاعات المجموع
- بقية مدخلات الأقسام (لا توجد أقسام أخرى)

مؤشر/علم الإقلاع (1 بايت)

خانة علم التنشيط. أول عنصر في مدخلة جدول الأقسام، يوجد فقط في أنظمة x86 ويشير إلى تنشط وحدة التحزين (أو العكس). تقنيا، يمكن تعيين هذا العلم في قسم أولي واحد فقط. علم الإقلاع بطول واحد بايت.(8 بت)، يستخدم منها بت واحد فقط (يتم تعيين البت الأكثر أهمية (10000000)_[58]، ليشكل هذا البايت، 80h)، قيم هذه الخانة تحمل احتمالين فقط؛ إما **80** وتشير إلى قسم قابل للإقلاع أو **00** غير قابل للإقلاع. شفرة الإقلاع سوف تقفز إلى قطاع الإقلاع في أول قسم قابل للإقلاع[27].

يمكن أن يكون هناك أكثر من نظام تشغيل مع نظام ملفات مختلف في أكثر من وحدة تخزين على القرص. مثلا، ويندوز في القسم الأولي الأول [45] وتوزيعة لينكس في القسم المنطقي الخامس.الخ. يمكن تحويل علم الإقلاع إلى قسم أولي أخر بتعديل هذا الحقل في جدول الأقسام باستخدام إحدى أدوات إدارة القرص مثل برنامج إدارة الأقراص في ويندوز 2000 أو FDISK في أنظمة دوس. بعض برامج إدارة الإقلاع تسمح بوجود القسم النشيط ضمن الأقسام المنطقية [33]، لكن مدير الإقلاع في الغالب لن يستخدم هذا المؤشر.وباستثناء **00h**، هذه القيمة نادرا في سجلات EBR [34] .

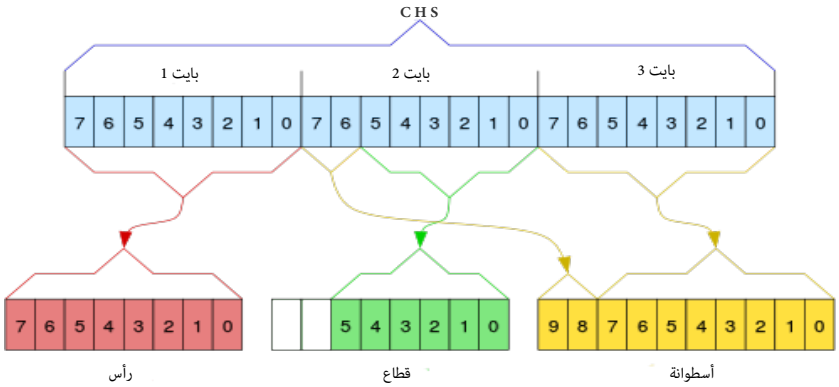
حقول قطاع البداية والنهاية (أسطوانات، رؤوس، قطاعات)

هذه الحقول في مدخلة القسم تعرف بمتتابعة CHS. ومطلوبة لبدء تشغيل الحاسوب في أنظمة x86. شفرة الإقلاع تستخدم هذه الحقول لإيجاد وتحميل قطاع إقلاع القسم النشط. [27]

- حقل أسطوانة البداية والنهاية، (كل واحد) 10-بت (8 بت +2 بت الأكثر أهمية MSB [58] من حقل القطاع)، نطاقها 1024 أسطوانة (من 0 إلى 1023) الممكن تعريفها في مدخلة القسم.
- حقول رأس البداية والنهاية (كل واحد) 1 بايت (8 بت)، نطاقها 256 (من 0 إلى 255). لكن عمليا هي 255 رأس (من 0 إلى 254) [47]
- حقول قطاع البداية والنهاية (كل واحد) 6 بت (6 بت الأقل أهمية LSB [58] في بايت الوسط)، نطاقها 63 (من 1 إلى 63).

لأن تعداد القطاعات يبدأ من 1 لذلك الرقم هو 63 في العدد الأقصى للقطاعات على المسار.

هذه 3 بايت عادة تكون بالترتيب رأس، ثم قطاع، ثم أسطوانة، لكن قيمة الأسطوانة تحتاج إلى أكثر من 8-بت وقيمة القطاع تستخدم أقل من 8-بت، وهذا سيحتاج إلى عملية تحويل كالتالي:



توزيع 24 بت على حقول CHS الثالثة في مدخلة القسم (أنظر جدول 7)

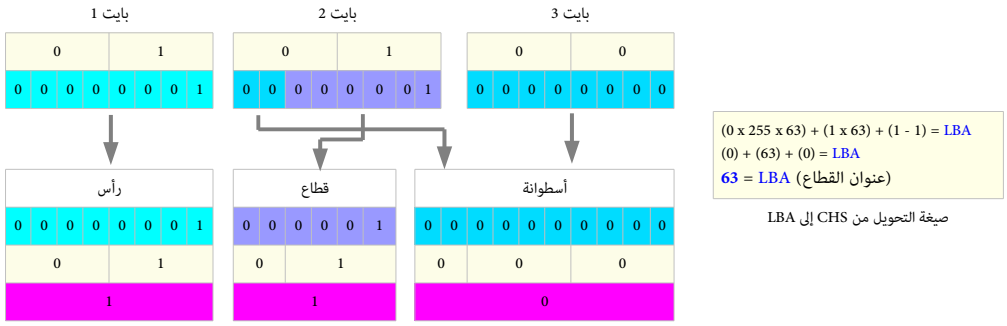
255 رأس (2^8-1)، [7] و 63 قطاع (2^6-1)، و 1024 أسطوانة (2^{10}) باستخدام امتدادات BIOS INT 13h

قطاع بداية القسم (3 بايت).

في المثال، القيم التالية 01 01 00 تشير إلى العنوان (**0**، **1**، **1**) CHS أو القطاع 63 في LBA (أي القطاع 64 على القرص، لأن الحساب يبدأ من **0** LBA)

هذه القيم في متتابعة CHS تحدد بدقة موقع أول قطاع في القسم [47]، **شرط** أن يكون ضمن أول **1024 أسطوانة** على القرص الثابت. إذا كان القطاع يتعد هذا الحد، متتابعة CHS ستكون في أقصى قيمها وهي: **1023**، **254**، **63** (التي تمثل الأسطوانة **1024**، الرأس **255**، القطاع **63**، لأن الأسطوانة والرأس تبدأ الحساب من **0**). وتظهر هنا في مدخلة القسم الثاني على شكل 3 بايت: FE FF FF . هذه القيم، حيلة من المبرمجين لإجبار أنظمة BIOS على استخدام نداءات LBA Int 13 عوض إحداثيات CHS. شفرة الإقلاع BIOS MBR سوف تتفحص معاملات CHS إذا وجدتها في أقصى قيمها، (1023*254*63)، تستخدم عوض ذلك، معاملات وعنونة LBA.. راجع شرح فك رموز هذه 3 بايت في قطاع نهاية القسم (أسفل). [31]

```
01B0                                                 00 01 [................]
01C0 01 00 DE FE 3F 04 3F 00 00 00 86 39 01 00 80 00 [..bb?.?...|9..|.]
01D0 01 05 07 FE FF FF C5 39 01 00 F8 AF 4E 09 00 00 [...byyA9..0 N...]
01E0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [................]
01F0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 55 AA [..............U.]
     00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F  0123456789ABCDEF
```



3 بايت تعطينا قيم قطاع بداية القسم في CHS. (نستخلص بتات بإستخدام النظام الثنائي)

(0 x 255 x 63) + (1 x 63) + (1 - 1) = LBA
(0) + (63) + (0) = LBA
63 = LBA (عنوان القطاع)

صيغة التحويل من CHS إلى LBA

قطاع نهاية القسم (3 بايت).

رغم سهولة حساب رقم الرأس (دائما قيمة +**1**) في بايت الأول، أرقام القطاع والأسطوانة في بايت الثاني والثالث، ترتيبها غير متجانس وتحتاج معالجة خاصة. الخطاطة التالية تماما مثل السابقة سوف تشرح فك رموز هذه 3 بايت.

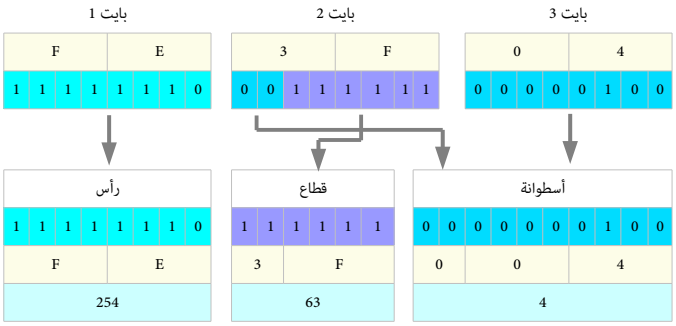
8 بت (**1111 1110**) في بايت الرأس تساوي في ست عشري **FE**، وفي العشري **254**. بما أن تعددها يبدأ مع رأس **0**، فهذا يشير إلى الرأس 255.

حساب قيمة القطاع يبدأ من أول **6 بت** (بدأ من بت الأقل أهمية، من **0 إلى 5**). بذلك بايت الثاني في ست عشري **3Fh** ينتج القطاع **3Fh** (نفس القيمة) وفي العشري **63**؛ من بتات **11 1111** .

قيمة الأسطوانة بطول **10 بت**، منها **2 بت الأكثر أهمية** [58] تأتي من بايت الثاني (**0011 1111**=3Fh)، مع الحفاظ على قيمة بايت الثالث **04h** في **8 بت الأدنى**. النتيجة ستكون: **0100 0000 00**، التي تعطينا قيمة الأسطوانة **04h** أو في العشري **4**. بذلك، متتابعة CHS في هذا المثال تمثل قطاع نهاية القسم بالشكل: (**63**، **254**، **4**).

إذا استخدمنا حاسبة خاصة مثل gcalctool مع هذه المتتالية FE 3F 04، التي في قطاع نهاية القسم في CHS، سنجد أن هذا القسم بحجم يكافئ التالي (مع طرح 63 LBA قطاع التي تسبق):

(255 × 63 × 5) = 80325 × 512 − **63** =. 41,094,144 بايت (تقريبا 40 ميغابايت) وهو نفس حجم القسم بعدد القطاعات في حساب مدخلة حجم القسم .



3 بايت تعطينا قيم قطاع نهاية القسم في CHS. (نستخلص بتات بإستخدام النظام الثنائي)

(4 × 255 × 63) + (254 × 63) + (63 − 1) = 80324
(64260) + (16002) + (62) = 80324
80324 = LBA (عنوان القطاع)

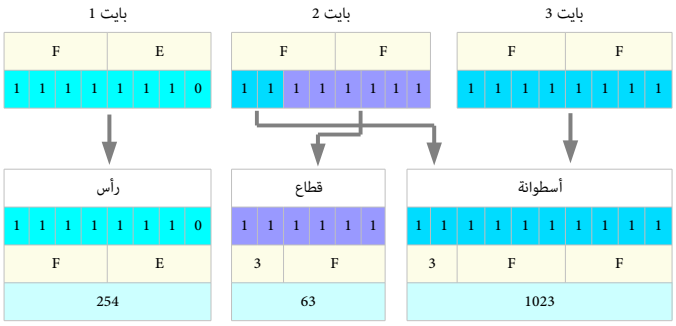
صيغة التحويل من CHS إلى LBA

- بايت الأول: **254** = **FE**h (هذا القرص يملك **255** رأس لكل أسطوانة).
- بايت الثاني: 63 = **3F**h ؛ (**0011 1111**) بت 6 وبت 7 بدون أهمية [58] (كلاهما صفر)؛ إذن القيمة هي **63** قطاع لكل رأس.
- بايت الثالث: 4 = **04**h (بتات من 4 إلى 7 بدون أهمية وبنقل بت 6 و 7 من بايت الثاني إلى خانة بت 8 وبت 9 : نحصل على الثنائية **00 0000 0100** إذن **4** هي قيمة الأسطوانة).

بالتالي، البيانات السابقة تماثل قيم قطاع النهاية **4**، **254**، **63** في متتابعة CHS (أي **5** أسطوانة، **255** رأس، و **63** قطاع).

كما ذكرنا سابقا، في الأقراص الكبيرة ستكون 3 بايت في متتابعة CHS دائما محشوة بالقيم FE FF FF (وتساوي 1024، 63، 255) في الأقسام التي تبدأ أو تنتهي خلف الأسطوانة 1024:

```
03be                                            80 20 |........KzU.... |
03c0 21 00 07 df 13 0c 00 08 00 00 00 20 03 00 00 df |!.......... ....|
03d0 14 0c 07 fe ff ff 00 28 03 00 00 98 00 0f 00 fe |.......(........|
03e0 ff ff 07 fe ff ff 00 c8 03 0f 00 98 00 0f 00 fe |................|
03f0 ff ff 0f fe ff ff 74 69 04 1e 8e 21 34 1c 55 aa |......ti...!4.U.|
     00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F  0123456789ABCDEF
```



3 بايت تعطينا أقصى قيم CHS وتعني أن القسم وراء 8.4 جيجابايت. (نستخلص البتات بإستخدام الثنائي)

(1023 × 255 × 63) + (254 × 63) + (63 − 1) = 16450559
(16434495) + (16002) + (62) = 16450559
16450559 = LBA (عنوان القطاع)

صيغة التحويل من CHS إلى LBA

- بايت الأول : FEh=254 لمجموع عدد الرؤوس 255.
- بايت الثاني والثالث: **FFh** و **FFh** تحلل إلى عددان ثنائيان كاملان من **6 بت** (**11 1111**) أي **3Fh**= **63** قطاع، و **10 بت** (**11 1111 1111**) أي **3FFh** =**1023** لمجموع **1024** أسطوانة, أي **63**، **254**، **1023**. هذه المتتابعة تكافئ **قطاع LBA** الذي هو بقيمة: **16450559**. إلى هذا الحد يمكن النفاذ إلى حوالي **8,4 جيجابايت** من قطاعات القرص الثابت.

مدخلات 16-بايت في جدول الأقسام لا يمكنها أن تتجوز **1024** أسطوانة ضمن متتابعة CHS في حقول بداية ونهاية القسم..

إذا استخدمنا حقول القيم الأقصى الممكنة، ستكون النتيجة السعة القصوى التي تقارب 8 جيجابايت. قبل استخدام ترجمة قياسات القرص عن طريق (واجهة) **امتدادات** BIOS INT 13h، كان حجم القسم الأولي النشيط لا يمكن أن يتجاوز 7.8 جيبيايت، بغض النظر عن نوع نظام الملفات المستخدم.

16,450,560 قطاع	×	**512** بايت لكل قطاع	=	**8,422,686,720** بايت	=	**8,4** جيجابايت أو 7.8 جيبايت [48]		
حجم قطاع	×	أسطوانات (10-بت)	×	رؤوس (8 بت)	×	قطاعات لكل مسار (6 بت)	=	سعة قصوى
sector size	×	cylinders	×	heads	×	sectors per track	=	MaxCapacity

للسماح بحجم أكبر من 7.8 جيبايت،[48] الأنظمة الآن تتجاهل قيم حقول قطاع البداية والنهاية في جدول الأقسام، وتستخدم مجموع القطاعات والقطاعات النسبية (المكافئة للعنوان LBA).

البرامج الخاصة يمكنها عرض متتابعة CHS مع قيمة أسطوانة أكبر من 1023، فقط بحساب قيم CHS المستعارة من حجم القسم أو قطاع البداية التي تستخدم 4-بايت (راجع الفقرة أسفل).

نوع القسم (1 بايت)

هوية النظام، عنصر آخر في مدخلة القسم غالبا ما يشير إلى نوع نظام الملفات المستخدم في تهيئة وحدة التخزين مثل FAT32، NTFS ...الخ، مع أو بدون استخدام خاصية الاستجابة للخطأ [35]. هذا الحقل يحدد أيضا هوية القسم الممتد (أن وجد) [32]. ويندوز 2000 يستخدم هذا الحقل لتحديد مشغلات العتاد في نظام الملفات التي يجب تحميلها أثناء بدء التشغيل.

أي أنه هناك عدة احتمالات في هذا الحقل، تقريبا 256 قيمة أغلبها يشير لأنظمة الملفات التي يمكن استخدمها في مخطط تقسيم القرص دوس، مثلا في ويندوز أن تي يستخدم نظام ملفات NTFS مع 07h بينما في أنظمة جنو/لينكس يستخدم 83h مع نظام ملفات EXT4 بالإضافة إلى 82h المستخدم في قسم الذاكرة الظاهرية (إبدال)...

0	Empty	1e	Hidden W95 FAT1	80	Old Minix	be	Solaris boot
1	FAT12	24	NEC DOS	81	Minix / old Lin	bf	Solaris
2	XENIX root	39	Plan 9	**82**	**Linux swap** / So	c1	DRDOS/sec (FAT-
3	XENIX usr	3c	PartitionMagic	**83**	**Linux**	c4	DRDOS/sec (FAT-
4	FAT16 <32M	40	Venix 80286	84	OS/2 hidden C:	c6	DRDOS/sec (FAT-
5	Extended	41	PPC PReP Boot	85	Linux extended	c7	Syrinx
6	FAT16	42	SFS	86	NTFS volume set	da	Non-FS data
7	**HPFS/NTFS**	4d	QNX4.x	87	NTFS volume set	db	CP/M / CTOS / .
8	AIX	4e	QNX4.x 2nd part	88	Linux plaintext	**de**	**Dell Utility**
9	AIX bootable	4f	QNX4.x 3rd part	8e	Linux LVM	df	BootIt
a	OS/2 Boot Manag	50	OnTrack DM	93	Amoeba	e1	DOS access
b	W95 FAT32	51	OnTrack DM6 Aux	94	Amoeba BBT	e3	DOS R/O
c	W95 FAT32 (LBA)	52	CP/M	9f	BSD/OS	e4	SpeedStor
e	W95 FAT16 (LBA)	53	OnTrack DM6 Aux	a0	IBM Thinkpad hi	eb	BeOS fs
f	W95 Ext'd (LBA)	54	OnTrackDM6	a5	FreeBSD	ee	EFI GPT
10	OPUS	55	EZ-Drive	a6	OpenBSD	ef	EFI (FAT-12/16/
11	Hidden FAT12	56	Golden Bow	a7	NeXTSTEP	f0	Linux/PA-RISC b
12	Compaq diagnost	5c	Priam Edisk	a8	Darwin UFS	f1	SpeedStor
14	Hidden FAT16 <3	61	SpeedStor	a9	NetBSD	f4	SpeedStor
16	Hidden FAT16	63	GNU HURD or Sys	ab	Darwin boot	f2	DOS secondary
17	Hidden HPFS/NTF	64	Novell Netware	b7	BSDI fs	fd	Linux raid auto
18	AST SmartSleep	65	Novell Netware	b8	BSDI swap	fe	LANstep
1b	Hidden W95 FAT3	70	DiskSecure Mult	bb	Boot Wizard hid	ff	BBT
1c	Hidden W95 FAT3	75	PC/IX				

لائحة FDISK

لمعلومات أكثر راجع كتيب:
نوع القسم في MBR/EBR
(تنبيه: أبحث عن نسخة الكتيب المراجعة)
راجع أيضا أنواع أقسام GPT في مسودة كتيب:
"جدول أقسام GUID"

حقل القطاعات النسبية وحقل قطاعات المجموع

القطاعات النسبية تكافؤ رقم قطاع بداية القسم باستخدام عنونة الكتل المنطقية (القطاع المطلق).

حقل القطاعات النسبية في مدخلات الأقسام الأولية، يمثل الحيد من بداية القرص إلى بداية وحدة التخزين (بمعنى: عدد القطاعات قبل القسم) [47]، أما حقل عدد القطاعات فيمثل العدد الإجمالي للقطاعات في وحدة التخزين (بمعنى: طول القسم). استخدام حقول القطاعات النسبية والمجموع ينتج عنها 32-بت أي توفر 8 بت إضافية مقارنة بمخطط 24-بت CHS [31] في تمثيل مجموع عدد القطاعات. هذا يسمح بتعريف أقسام يصل عدد قطاعاتها إلى 2^{32} قطاع أو 4,294,967,296 بايت.

في ويندوز أن تي استخدمت حقول CHS فقط للتوافق مع الإصدارات السابقة؛ مثل م.س.دوس و ويندوز 95، وللتوافق عند بدأ التشغيل مع BIOS INT 13.

مع حجم القطاع المعياري 512 بايت، تستخدم قيم 32-بت في تمثيل حقول القطاعات النسبية وقطاعات المجموع وتترجم إلى الحجم الأقصى للقسم 2 تيرابايت أو 2,199,023,255,552 بايت. تخطيط العنونة هذا يستخدم في أنظمة ويندوز مع نظام ملفات NTFS و FAT32 فقط.

استخدام حقول القطاعات النسبية والمجموع مع الأقراص المنطقية [33] في القسم الممتد [32] يختلف عن استخدامها في الأقسام الأولية. بالنسبة للمدخلة 1 في القسم لكل قرص منطقي، حقل القطاعات النسبية سيكون الحيد من القطاع بداية القرص المنطقي [47] الذي يتضمن قطاع إقلاع القسم. وحقل قطاعات المجموع يمثل عدد القطاعات من قطاع إقلاع القسم إلى نهاية القرص المنطقي. بالنسبة للمدخلة 2 في القسم، حقل القطاعات النسبية سيكون الحيد من بداية القسم الممتد إلى القطاع الذي يتضمن جدول أقسام القرص المنطقي المحدد في المدخلة 2 في القسم. وحقل قطاعات المجموع يمثل الحجم الإجمالي للقرص المنطقي المحدد في المدخلة 2 في القسم (راجع أيضا كتيب EBR).

```
01B0                                                    00 01 [................]
01C0 01 00 DE FE 3F 04 3F 00 00 00 86 39 01 00 80 00 [..bb?.?...|9..|.]
01D0 01 05 07 FE FF FF C5 39 01 00 F8 AF 4E 09 00 00 [...byyA9..0 N...]
01E0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [................]
01F0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 55 AA [..............U.]
     00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F  0123456789ABCDEF
```

قطاع البداية (4 بايت)

هذه القيمة تشير إلى أول قطاع من القسم، (نفس رقم القطاع الذي تقدمه قيم البداية في عنونة CHS) [31]. لكن هذا الحقل بطول 4-بايت ويستخدم عنونة LBA (والحساب يبدأ من القطاع المطلق **0**)، هذا يعني إمكانية تحديد بداية القسم ضمن أول **4,294,967,295** قطاع على القرص الثابت، أي في حدود **FFFF FFFFh** التي تساوي **2 تيرابايت** [46].

بعد الحصول على قيمة **4** بايت نحتاج إلى عكس ترتيب الثمانيات (تنبيه: هذا ترتيب بايت وليس ترتيب ست عشري، لأنها ستكون مخزنة على القرص بترتيب نهوي-صغير).

في المثال كانت: **3F 00 00 00** بعد عكس ترتيبها حصلنا على **3F** والتي تعني أن هذا القسم الأول يبدأ عند القطاع المطلق **63** LBA (أو القطاع 64 على القرص). وهذا هو قطاع الإقلاع الأول الممكن على أي قرص يملك 63 قطاع لكل رأس/مسار. (ملاحظة: معظم سجلات إقلاع القسم، حتى وقت قريب، كانت تبدأ عند القطاع المطلق 63.). [47]

حجم القسم (4 بايت).

تشير إلى إجمالي عدد القطاعات، هذه 4 بايت أيضا تسمح بحجم قسم يصل إلى **2 تيرابايت**، وهي كذلك مخزنة على القرص بترتيب نهوي-صغير. في المثال كانت: 00 01 39 86

بعد عكس ترتيبها من نهوي صغير (ترتيب بايت وليس ترتيب ست عشري) حصلنا على 13986 ست عشرية أي 80262 قطاع أو تقريبا 40 ميغابايت (**80262** × 512 = 41094144 بايت).

حساب حجم القسم بشكل مباشر

(0 x 255 x 63) + (1 x 63) + (1 - 1) = **63**
(4 × 255 × 63) + (254 × 63) + (63 − 1) = **80324**
80262 = (**80324** - **63**) + 1

عنوان قطاع النهاية LBA ناقص عنوان قطاع البداية LBA + **1**

> لا تغير إعدادات LBA على أي قرص ثابت يحتوي بيانات. لأنك سوف تأثر على عملية ترجمة النظام لخصائص القرص في تخزين البيانات وسوف تتلف جميع الملفات والأقسام على القرص الفيزيائي. راجع دليل الحاسوب، قبل تعديل إعدادات نظام BIOS.

إقلاع النظام

في الأجهزة المتوافقة مع أنظمة IBM، البرنامج الثابت للإقلاع المضمن في رقاقة/شريحة الذاكرة [20] BIOS ROM، سيكون المسؤول عن تحميل وتنفيذ سجل الإقلاع الرئيسي. [24]

وكي تظل متوافقة، جميع أنظمة x86، تبدأ مع معالج المكرو في نمط للتشغيل يدعى النمط الحقيقي.[38] حاسوب PC/XT-5160 الذي كان أول من وظف MBR، كان يستخدم معالج 8088.

نظام BIOS يقرأ MBR من جهاز التخزين في الذاكرة، ثم يوجه معالج المكرو إلى بداية شفرة الإقلاع. وبما أن BIOS يشتغل في نمط للعنونة حقيقي [38]، سيكون المعالج كذلك في ذلك النمط عندما يبدأ تنفيذ برنامج [21] MBR، ولذلك يتوقع أن تتضمن بداية MBR على تعليمات للغة الآلة في النمط الحقيقي. [24]

نظراً لصغر حجم الجزء الذي تحتله شفرة MBR، عادة، يتضمن فقط برنامج صغير [44] وظيفته نسخ شفرة إضافية (مثل محمل الإقلاع) من جهاز التخزين إلى الذاكرة. ثم ينتقل التحكم إلى تلك الشفرة المسؤولة عن تحميل نظام التشغيل الفعلي. هذه العملية تعرف باسم: chain loading وتعني "استبدال برنامج قيد التنفيذ ببرنامج جديد".

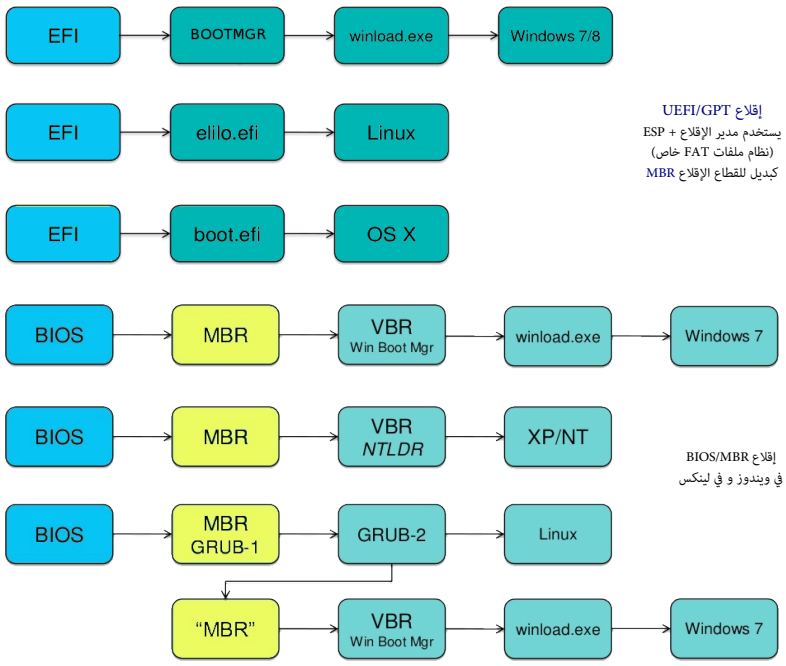
العديد من برامج شفرة MBR التي كانت معروفة سابقا، نشأت أصلا لإقلاع أنظمة PC DOS و MS-DOS، هذا النوع من الشفرات استمر استخدمها بعد ذلك على نطاق واسع. قطاعات الإقلاع تلك كانت تعتمد على استخدام مخطط جدول أقسام في برنامج fdisk، الذي يتفحص جدول أقسام MBR لإيجاد القسم النشيط.[25] [27] ثم يحمل ويشغل شفرة VBR في القسم النشيط.

هناك عدة تطبيقات بديلة لشفرة الإقلاع [44]، بعضها ينصب بواسطة مدير الإقلاع وتعمل بطرق مختلفة، بعض شفرات MBR تحمل وتنفذ شفرة إضافية لمدير الإقلاع من أول مسار على القرص.

برامج MBR يمكنها التفاعل مع المستخدم في تحديد قسم وقرص الإقلاع، وقد ينقل التحكم إلى MBR في قرص آخر. بعض شفرات MBR يمكن أن تتضمن لائحة بمواقع القرص (غالبا ما تشير إلى ملفات داخل نظام ملفات) تتضمن بقية شفرة مدير الإقلاع كي يتم تحميلها وتنفيذها. (الطريقة الأولى لا تستخدمها جميع برمجيات تقسيم القرص، خصوصا تلك التي تقرا وتكتب إلى GPT. والطريقة الأخيرة تتطلب تحديث لائحة مواقع القرص المضمنة عند إجراء أية تغيرات من شأنها تغيير مكان الشفرة).

في الأجهزة التي لا تستخدم معالجات x86 ، أو في الأجهزة التي تستخدم تلك المعالجات لكن لا تملك البرنامج الثابت BIOS مثل OpenBoot أو واجهة البرنامج الثابت الممتد EFI، هذا التصميم غير مناسب، و MBR ليس جزء من عملية الإقلاع. [26] في هذه الحالات، يستخدم البرنامج الثابت EFI الذي يفهم مخطط أقسام GPT، ويوظف نظام ملفات FAT، ويستطيع تحميل وتشغيل ملفات البرامج المضمنة في قسم الإقلاع ESP [27]، في هذه الحالة يتم تضمين MBR فقط للتوافق مع برمجيات القرص القديمة التي لا تفهم GPT. (راجع: كتيب "جدول أقسام "GUID")

نظام BIOS لا يستطيع الاقلاع مباشرة من أقراص GPT ، لكن هناك بعض الشفرات البديلة لسجل MBR، تحاكي إقلاع البرنامج الثابت EFI، وتسمح للأجهزة التي لا تملك رقاقة EFI الإقلاع من أقراص GPT. هذه الشفرات يمكنها اكتشاف GPT، ووضع المعالج في نمط التشغيل الصحيح، وتحميل الشفرة المتوافقة مع نظام EFI من القرص.



أسماء برامج وملفات محملات الإقلاع المختلفة ←	GNU GRUB 2, BOOTMGR, winload.exe, NTLDR, elilo.efi (LILO تنويعة ليلو), boot.efi (ماك أو.إس)
مرحلة الإقلاع الأولي [44] (شفرة محمل الإقلاع جزء من MBR) ←	GRUB-1 = GRUB stage 1
مرحلة الإقلاع الثانية (شفرة محمل الإقلاع غالبا توجد على المسار 0 بعد MBR أو في نظام الملفات) ←	GRUB-2 = GRUB stage 2

لمعلومات أكثر عن محمل الإقلاع في لينكس، راجع الدليل الرسمي للبرنامج GRUB 2 في الموقع الرسمي والشفرة الأصلية في موقع الأرشيف.

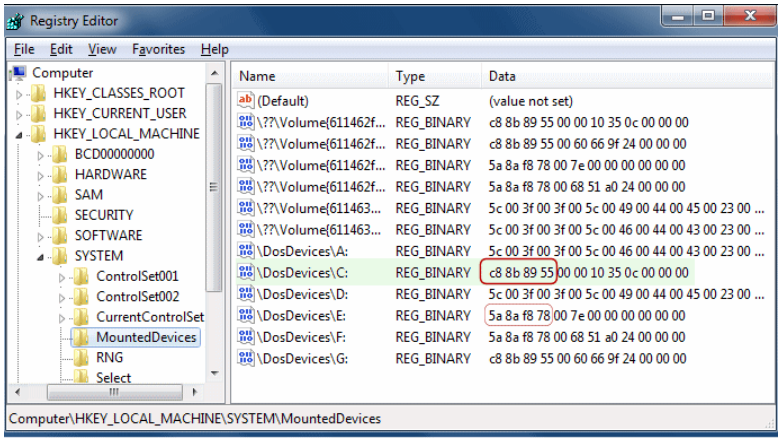
توقيع القرص

(هوية القرص، 4 بايت اختيارية للتعريف بالقرص تبدأ عند الحيد 1B8h، وقد تكون أكثر من أربعة بايت في أنظمة معينة) .

بالإضافة إلى [21] [44] شفرة الإقلاع وجدول الأقسام، سجل MBR يمكن أن يتضمن أيضا على توقيع للقرص [36]. وهو عبارة عن قيمة 32-بت المقصود منها إعطاء تعريف فريد للقرص.

توقيع القرص تم طرحه أول مرة في نظام ويندوز آن تي 3.5، لكنه الآن يستخدم في العديد من أنظمة التشغيل، بما فيها إصدارة نواة لينكس 2.6 والأحدث. أدوات جنو/لينكس يمكنها استخدام توقيع قرص آن تي في تحديد قرص الإقلاع.[28]

ويندوز آن تي (والأنظمة اللاحقة) تخزن توقيع القرص [36] عند بداية الحيد 1B8h. وتستخدمه كمؤشر لجميع الأقسام على أي قرص متصل بالحاسوب تحت النظام؛ هذه التواقيع تحفظ في مفاتيح مسجل نظام ويندوز، أصلا، من أجل تخزين قيم التعيين بين الأقسام ومحارف الأقراص. و أيضا يمكن استخدامها في ملفات BOOT.INI (رغم أن ذلك لا يحدث غالبا)، لوصف مواقع أقسام ويندوز آن تي (والأنظمة اللاحقة) التي تقبل الإقلاع. [29] المثال التالي سيكون عن أحد مفاتيح توقيع قرص آن تي في مسجل ويندوز XP/ 2000:



HKEY_LOCAL_MACHINE\SYSTEM\MountedDevices\

إذا كان توقيع القرص المخزن في MBR بهذا الشكل والترتيب: A8h E1h B9h D2h وكان قسمه الأول في ويندوز مقترن بالقرص المنطقي C: [45] حينذاك ستكون بيانات REG_BINARY في المسجل تحت قيمة المفتاح \DosDevices\C: بهذا الشكل:

A8 E1 B9 D2 00 7E 00 00 00 00 00 00

4 بايت الأولى هي توقيع القرص. [36] (لحظ: في المفاتيح الأخرى، هذه بايتات قد تظهر بترتيب معكوس، مثل بايتات قطاع MBR) هذه بايتات يتبعها 8 بايت أخرى، تشكل عدد صحيح 64-بت، بترتيب نهوي صغير، تستخدم في تحديد موقع إزاحة البايت لهذا القسم. في هذه الحالة، 00h 7Eh تشير للقيمة الست عشرية 7E00h أو 32,256 العشرية.

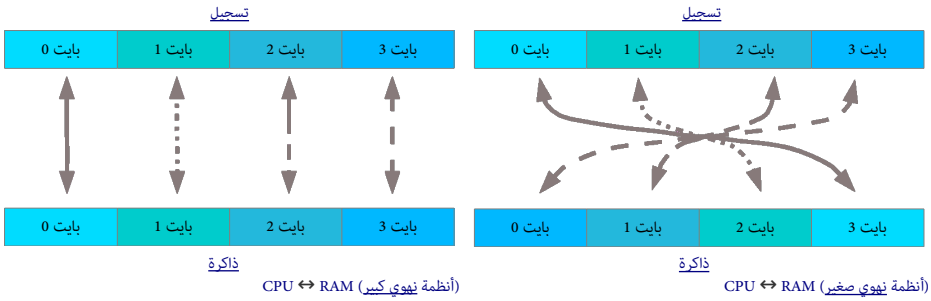
لنفترض أن هذا الجهاز عرض قيمة للقطاع 512 بايت، إذا قسمنا إزاحة البايت هذه على 512 بايت، (32256 ÷ 512) الناتج 63 سيكون رقم القطاع الفيزيائي (أو LBA) الذي يتضمن القطاع الأول في القسم [47] (تنبيه: على عكس حساب القطاع المستخدم في قيمة قطاعات متتابعات CHS، التي تبدأ التعداد من واحد، قيمة القطاع المطلق أو LBA تبدأ التعداد من الصفر).

إذا كان هذا القرص يملك قسم أخر مع القيم 00h F8h 93h 71h 02h تتبع توقيع القرص (مثلا، تحت قيمة المفتاح \DosDevices\D:)، سيبدأ عند إزاحة البايت 00027193F800h أو 10,495,457,280 بالعشري، إذا قسمنا (10,495,457,280 ÷ 512) الناتج سيكون أيضا البايت الأولى للقطاع الفيزيائي 20,498,940.

منذ صدور ويندوز فيستا أصبح توقيع القرص يخزن أيضا في ذاكرة بيانات تضبيط الإقلاع BCD وتعتمد عليه عملية الإقلاع.[30] أي إذا تغير توقيع القرص، أو لا يمكن إيجاده، أو تعارض مع توقيع أخر، سيكون ويندوز غير قادر على الإقلاع. [31] إلا إذا أجبر ويندوز على استعمال الجزء المركب من عنوان LBA في المدخلة AAP.كتوقيع قرص مستعار، هذا الاستخدام في ويندوز يتعارض مع وظيفة AAP [30] في أنظمة PTS-DOS 7 و DR-DOS 7.07، خصوصا إذا كانت شفرة إقلاع الأنظمة تقع خارج 8 جيجابايت الأولى من القرص، ولذلك يجب استعمال عنونة LBA [31].

اعتبارات برمجية

بسبب نشأت MBR في أنظمة PC XT [32] نجد أن الأجهزة المتوافقة مع أنظمة IBM تستخدم نهوي صغير في ترتيب البيانات، هذا يعني أن المعالج يخزن، القيم الرقمية التي تغطي 2 بايت أو أكثر، بترتيب عكسي في الذاكرة، أي سيكون البايت الأدنى أولا [58] LSB، هذه القاعدة تعكسها بنية MBR على الوسط. لذلك توقيع الإقلاع يظهر في محرر القرص بهذا الترتيب: 55h AAh [1]



آلية الإقلاع في BIOS [20] سوف تبدأ بتحميل أول MBR صالح تجده في الذاكرة عند العنوان 0000h :7C00h [32] [39] أخر تعليمة تنفذ في شفرة BIOS ستكون "قفزة" إلى ذلك العنوان، لتوجيه التنفيذ إلى بداية نسخة MBR . معظم أنظمة BIOS تتحقق أولا من التوقيع عند الإزاحة 1FEh+، رغم أن تطبيق BIOS قد يختار عمل فحوص إضافية أخرى، مثل التأكد من احتواء MBR على جدول أقسام صالح بدون أية مدخلات تشير إلى قطاعات تتجاوز سعة القرص المعلن عنها.

يتوقع تحميل شفرة قطاع إقلاع MBR عند العنوان الفيزيائي 0000h:7C00h، [8]، وستكون كامل الذاكرة متوفرة في النمط الحقيقي [38]؛ من العنوان 0000h:0501h (العنوان 0000h:0500h هو أخر عنوان يستخدمه Phoenix BIOS) [14] إلى العنوان 0000h:7FFFh. [32] الذي امتد لاحقا إلى 0000h:FFFFh [33] (وأحيانا [9] إلى (9000h:FFFFh - نهاية أول 640 كيلوبايت - [10]

نداء المقاطعة INT 12h يمكن أن يساعد في تحديد حجم الذاكرة الذي يمكن تخصيصه بأمان (عادة، يقرأ حجم الذاكرة الأساسة [41] بالكيلوبايت من موقع 0040h:0013h أي العنوان **الخطي** segment:offset، لكن نداء المقاطعة [55] هذا يمكن أن يكون محتجز [54] من قبل برمجية إقلاع تمهيدية أخرى مقيمة في الذاكرة مثل أغطية نظام البيوس[57] BIOS overlays، أو شفرة RPL أو حتى فيروسات [51] كي تقلل من الكمية المعلن عنها من الذاكرة المتوفرة حتى تمنع برمجيات مرحلة الإقلاع الأخرى مثل قطاعات الإقلاع من إعادة كتابتها).

أخر 66 بايت في MBR محجوزة من أجل جدول الأقسام [26] ومعلومات أخرى، لذا يجب أن يكون برنامج قطاع إقلاع MBR صغير بما فيه الكفاية كي يتناسب مع 446 بايت أو أقل في الذاكرة. شفرة MBR يمكنها التواصل مع المستخدم وتفحص جدول الأقسام. لكن مهمتها الرئيسية تحميل البرنامج الذي سيكمل تنفيذ مرحلة الإقلاع التالية، عادة، بتوظيف INT 13h. لكن MBR أو أجزاء منه، [11] قد يبقى في الذاكرة ويستخدم كجزء من البرنامج المحمل، بعد أن ينقل MBR التحكم إلى ذلك البرنامج. نفس الشيء ينطبق على VBR، سواء كان قرص مرن أو قسم على قرص ثابت.[27]

لكن عادة البرنامج المحمل بواسطة برنامج سجل الإقلاع يقوم بطرح وإعادة كتابة صورة RAM لهذا الأخير، لذلك وظيفيا البرنامج هو كحلقة أولى فقط في سلسلة محمل الإقلاع.

من وجهة نظر تقنية، الاختلاف بين MBR و VBR موجود فقط على مستوى برمجيات المستخدم، أي فوق البرنامج الثابت BIOS. (هنا "برمجيات المستخدم" تعني برامج نظام التشغيل والتطبيقات) بالنسبة لنظام BIOS، لا فرق بين الأقراص القابلة للإزالة (مثل، القرص المرن) والأقراص الثابتة. ففي كلاهما، BIOS يقرأ القطاع الفيزيائي الأول من الوسيط في الذاكرة RAM عند العنوان المطلق 7C00h، ويتحقق من التوقيع في أخر 2 بايت من القطاع المحمل، ثم إذا عثر على التوقيع الصحيح، ينقل التحكم إلى أول بايت في القطاع مع تعليمة القفزة (JMP). الفرق الوحيد الحقيقي عند نظام BIOS هو أنه (افتراضيا، أو إذا كان ترتيب الإقلاع لا يمكن ضبطه) يحاول الإقلاع من أول قرص قابل للإزالة قبل أن يحاول الإقلاع من القرص الثابت الأول.

من منظور BIOS، عملية تحميل MBR [21] لسجل VBR في ذاكرة RAM هي تماما مثل عملية تحميل VBR في القرص المرن للشفرة الكائن (تدعى أيضا: الكود الغرضي/الكود النهائي) الخاصة بمحمل نظام التشغيل في ذاكرة RAM. ففي كلتا الحالتين، البرنامج الذي يحمله BIOS سوف يعمل على تحميل نظام التشغيل عن طريق آلية chain loading.

الفرق بين السجلان MBR و VBR تجريدي على مستوى برمجية نظام التشغيل، ومصمم لمساعدة المستخدم على فهم التنظيم الوظيفي وبنية النظام. هذا الفرق ليس موجود بالنسبة لنظام BIOS. سواء حمل مباشرة MBR أو VBR، سوف ينقل إلي ذلك السجل، التحكم الكامل في النظام، ثم يصبح BIOS في خدمة فقط ذلك البرنامج. ويصبح الجهاز تحت سيطرة البرنامج المحمل (على الأقل، حتى إعادة الإقلاع التالي). بعد حصوله على التحكم الكامل، هذا البرنامج لا يحتاج إلى الاتصال مرة أخرى بنظام BIOS وربما يغلق هذا الأخير كليا، عن طريق إزالة متجهات روتين المقاطعة ISR BIOS من جدول متجهات المقاطعة IVT [56]، ثم إعادة كتابة منطقة بيانات السوس BDA.

ذكرنا هذا للتأكيد أن برنامج الإقلاع المحمل والمشغل من القطاع الأول على القرص من قبل BIOS يمكن أن يفعل أي شيء، طالما أن البرنامج لا يتصل بخدمات BIOS أو يسمح باستدعاء متجهات BIOS ISR بعد أن عطل سلطة BIOS اللازمة لعمل تلك الخدمات والمتجهات ISRs بالشكل الصحيح.

وكما ذكرنا أعلاه، شفرة الإقلاع التقليدية [44] [21] MBR تحمل وتشغل شفرة VBR (أيا كانت) من بداية القسم النشط. سجل إقلاع القسم التقليدي سيكون بحجم قطاع 512 بايت، لكن شفرة MBR يمكنها تحميل قطاعات إضافية للمحملات الإقلاع الأطول من قطاع واحد، رغم أنها لا تقوم بأي حسابات لمعرفة حجم القطاع، في الواقع، هناك على الأقل 1 كيلوبايت من ذاكرة RAM متوفرة عند العنوان 7C00h في جميع أجهزة IBM XT و AT، لذا قطاع من 1 كيلوبايت يمكن استخدمه من دون أية مشاكل. ومثل MBR يتوقع كذلك تحميل VBR عند عنوان الذاكرة 0000h:7C00h. هذا يعود إلى طبيعة تصميم VBR الذي نشأ في الوسط بدون تقسيم، حيث يحمل VBR مباشرة من قبل روتين إقلاع BIOS؛ وكما ذكرنا سابقا، نظام BIOS يتعامل مع سجلات MBR و VBR بنفس الطريقة [12]. بما أن هذا هو نفسه الموقع الذي يتم تحميل فيه MBR، أحد المهام الأولى لسجل MBR ستكون التحول من ذلك الموقع إلى مكان آخر في الذاكرة، يحدده MBR بنفسه، لكن غالبا سيكون 0000h:0600h (في أنظمة MS-DOS/PC DOS/OS/2 وفي شفرة MBR في ويندوز). أو سيكون 0060h:0000h (في معظم أنظمة DR-DOS).

(رغم أن هذان العنوانان للتقطيع كلهما يحددان نفس عنوان الذاكرة الفيزيائي [39] في نمط الحقيقي للمعالج [38]، نظام Apple Darwin، حتى يستطيع الإقلاع، يجب أن يتحول MBR إلى عنوان 0000h:0600h وليس 0060h:0000h. لأن الشفرة تعتمد على مؤشر DS:SI إلى مدخلة القسم الذي يوفره MBR، لكنه يشير إليه بالخطأ بواسطة فقط 0000h:SI [34] وعند إعادة تموضع شفرة MBR في مكان جديد، لا ينبغي أن يكون ذلك في عناوين أخرى في الذاكرة. لأن قطاعات إقلاع VBR كثيرة تفترض تخطيط معياري للذاكرة [41] عند تحميلها لملف إقلاعها.

حقل الحالة في مدخلة جدول الأقسام يستخدم للإشارة إلى القسم النشط. سجلات MBRs المتوافقة معياريا تسمح بقسم نشيط واحد فقط وتستخدم هذا كجزء من عملية التحقق، للتأكد من وجود جدول أقسام صالح.

وتعرض رسالة خطأ، إذا كان هناك أكثر من قسم يحمل علامة نشط. بعض سجلات MBRs الغير معيارية لا تعتبر هذا خطأ، وتستخدم أول قسم نشط من تلك الأقسام.

تقليديا، باستثناء قيمة 00h (غير نشيط) وقيمة 80h (نشيط) القيم الأخرى تعتبر باطلة وبرنامج الإقلاع سوف يعرض رسالة خطأ عندما يصادفها. لكن مواصفة القبس والتشغيل PnP BIOS ومواصفة إقلاع نظام البيوس BBS تسمح كذلك بالإقلاع من الأجهزة الأخرى منذ عام 1994.[33][35] ومنذ قدوم أنظمة MS-DOS 7.10 (ويندوز 95B) والأنظمة اللاحقة، بدأ MBR يتعامل مع تعيين بت 7 كعلم تنشيط ويعرض رسالة خطأ فقط مع القيم 01h....7Fh . واستمر يتعامل مع المدخلة كوحدة قرص فيزيائي تستخدم عند تحميل قطاع إقلاع VBR القسم المقابل فيما بعد، ولهذا تعتبر الآن أقراص الإقلاع الأخرى التي لا تستخدم 80h صالحة أيضا، لكن مايكروسوفت دوس لم يوظف هذه الاضافة.

تخزين رقم محرك الأقراص الفيزيائي الفعلي في جدول الأقسام عادة لا يسبب مشاكل في التوافق مع الإصدارات السابقة، لأن القيمة ستكون مختلفة عن 80h فقط على الأقراص الأخرى وليس القرص الأولى (التي لم تكن قابلة للإقلاع، بأية حال). حتى مع الأنظمة التي يمكنها الإقلاع من الأقراص الأخرى، هذه الإضافة ما زالت لا تعمل في العموم، مثلا، بعد تغيير قيم تعيين الأقراص الفيزيائية في BIOS بسبب إزالة أو إضافة أو تبديل الأقراص، ولذلك، وفقا لمواصفة إقلاع نظام البيوس BBS [33] أفضل طريقة في سجلات MBR الحديثة، قبول بت 7 كعلم تنشيط لتمرير قيمة DL التي يوفرها في الأصل BIOS بدلا من مدخلة جدول الأقسام.

واجهة BIOS ↔ MBR

سجل MBR سيحمل في الذاكرة عند موقع 0000h:7C00h مع تنصيب تسجيلات المعالج التالية عندما يقوم محمل الإقلاع الابتدائي (عادة، IPL [44] في أنظمة BIOS) بتمرير عملية التنفيذ إليه عن طريق القفز إلى العنوان 0000h:7C00h في النمط الحقيقي للمعالج [38]:

- CS:IP = 0000h:7C00h (ثابتة)

بعض أنظمة BIOS في أجهزة كومباك تستخدم بالخطأ العنوان 07C0h:0000h. رغم أن هذا العنوان يحدد نفس الموقع في ذاكرة النمط الحقيقي، إلا أنه غير معياري، ويجب تجنبه، فقد لا تعمل شفرة MBR التي تفترض قيم تسجيل معينة أو لم تكتب كي تنقل إلى مكان آخر.

- DL = وحدة قرص الإقلاع (رقم جهاز الإقلاع) [64]

في الأقراص المثبتة / الأقراص القابلة للإزالة: القرص الأول = 80h ، القرص الثاني = 81h ....إلى FEh

في الأقراص المرنة / تنويعات القرص [61] superfloppies : القرص الأول = 00h ، القرص الثاني = 01h ... إلى 7Eh

قيم 7Fh و FFh محجوزة من أجل الأقراص عن بعد / أقراص ROM، ويجب ألا تستخدم على القرص.

أنظمة IBM BIOS مثلها مثل معظم أنظمة BIOS الأخرى تدعم DL. لكن نظام توشيبا Toshiba T1000 BIOS معروف أنه لا يدعم هذا بالشكل الصحيح، بعض أنظمة Wyse 286 BIOS القديمة تستخدم قيم DL أكبر أو تساوي 2 مع الأقراص الثابتة (مما يجعلها تعكس أرقام الأقراص المنطقية تحت نظام دوس بدلا من أرقام الأقراص الفيزيائية في BIOS). عادة، USB sticks التي تم إعدادها كأقراص قابلة للإزالة تحصل على قيم الإسناد DL = 81h, 80h .. إلى آخره. لكن، بعض أنظمة BIOS النادرة تعرضها بالخطأ تحت DL = 01h، تماما كما لو تم إعدادها كأقراص [61] superfloppies.

أنظمة BIOS المعيارية والمتوافقة تخصص أرقام أكبر أو تساوي 80h خصيصا للقرص الثابت/ الأقراص القابلة للإزالة، وتقليديا فقط القيم 80h و 00h يتم تمريرها كوحدات أقراص فيزيائية أثناء الإقلاع. رسميا يتم تقسيم فقط الأقراص الثابتة / الأقراص القابلة للإزالة، لذلك قيمة DL الوحيدة التي يستطيع تقليديا رؤيتها MBR هي 80h. العديد من سجلات MBR مبرمجة لتتجاهل قيمة DL وتعمل مع القيمة المضمنة في الشفرة [24] (عادة تكون 80h) على أية حال. مواصفة القبس والتشغيل PnP BIOS ومواصفة إقلاع نظام البيوس BBS منذ 1994، تسمح أيضا بالإقلاع من الأجهزة الأخرى. [33][35]. المواصفة الأخيرة توصي باستخدام قيمة DL من قبل شفرات MBR و VBR بدلا من القيم الاعتيادية المضمنة داخليا [24]. [33] هذا سيضمن أيضا التوافق مع مختلف ألإسنادات الغير معيارية (انظر الأمثلة أعلاه)، طالما أنها متعلقة بشفرة MBR.

الأقراص المدمجة التي تقبل الإقلاع وتتبع مواصفة El Torito يمكنها أن تتضمن صور للقرص يصلها نظام BIOS لتظهر على هذه الواجهة كأقراص مرنة أو **[61]** superfloppies. قيم 01h و 00h في التسجيل DL يمكن استخدامها أيضا من قبل إضافات BIOS الخاصة بخدمات PARTIES، و TCG في وضعية مؤمنة للنفاذ إلى ما يدعى أقسام PARTIES الخفية، و ملفات صور القرص التي يحدد موضعها عبر سجل BEER في القطاع الفيزيائي الأخير من منطقة HPA على القرص الثابت. ورغم أنها مصممة لمحاكاة الأقراص المرنة أو تنويعات قرص [61] superfloppies، شفرة MBR التي تقبل هذه القيم الغير معيارية في DL تسمح باستخدام صور الوسيط المقسم على الأقل في مرحلة إقلاع نظام التشغيل.

- DH : بت 5 = 0

الجهاز المدعوم في BIOS من خلال INT 13h؛ عدا ذاك: لا يهم (يجب أن تكون صفر) **التسجيل** DH تدعمه بعض أنظمة IBM BIOS.

- بعض التسجيلات الأخرى في الأنظمة الأصلية IBM ROM BIOS قد تحتوي أيضا على قيم معينة (DS, ES, SS = 0000h; SP = 0400h)، لكن هذه القيم لا يعتمد عليها، لأن أنظمة BIOS الأخرى قد تستخدم قيم مختلفة. لهذا السبب، شفرة MBR من شركات مثل أي بي أم، ومايكروسوفت، ديجيتال ري سيرش،.الخ. لم تستفد أبدا من هذه الميزة. أيضا الاعتماد على قيم التسجيل هذه في قطاعات الإقلاع قد يسبب مشاكل في عمليات إقلاع chain-boot..

الأنظمة التي تدعم تقنية القبس والتشغيل PnP BIOS أو مواصفة إقلاع نظام البيوس BBS سوف توفر مؤشر إلى بيانات PnP بالإضافة إلى التسجيل DL : [33][35]

- DL = وحدة قرص الإقلاع (انظر أعلاه)
- ES:DI = تشير إلى بنية تفحص تنصيب "$PnP"

هذه المعلومات تسمح لمحمل الإقلاع في MBR (أو VBR في حالة التمرير) التفاعل مع BIOS أو [57] PnP / BBS BIOS overlay المقيم في الذاكرة لضبط ترتيب الإقلاع..الخ.، لكن هذه المعلومات يتم تجاهلها من قبل معظم سجلات MBR و VBR المعيارية. التسجيلان ES:DI يتم تمريرها بالكامل إلى VBR لاستخدامها لاحقا من قبل نظام التشغيل المحمل، لكن أنظمة التشغيل التي تمكن تقنية القبس والتشغيل PnP عادة تملك أيضا طرق احتياطية للاسترداد مدخلة PnP BIOS فيما بعد، لذا معظم أنظمة التشغيل لا تعتمد على هذا.

واجهة MBR ↔ VBR

سجل MBR المعياري [21] سينقل عملية التنفيذ إلى VBR المحمل بنجاح عند 0000h:7C00h، بالقفز إلى ذلك العنوان في النمط الحقيقي [38] مع الحفاظ على التسجيلات التالية أو إنشاءها:

- CS:IP = 0000h:7C00h (ثابتة) [13]
- DL = وحدة قرص الإقلاع (انظر أعلاه)

في أنظمة PC DOS 2.0-6.3 / MS-DOS 2.0-7.0، سجلات MBR لا تمرر قيمة DL المستلمة في المدخلة، ولكن تستخدم خانة حالة الإقلاع في مدخلة جدول الأقسام الخاصة بالقسم الأولي الذي اختير كوحدة جهاز إقلاع. وبما أنها ستكون نفس قيمة 80h الموجودة في معظم جداول أقسام MBR، فلن تغير شيء إلا إذا حاول BIOS إقلاع جهاز أخر غير القرص الأول في ترتيب الأقراص المشتة / الأقراص القابلة للإزالة. هذا أيضا يفسر لماذا تلك الأنظمة لا يمكنها الإقلاع من القرص الثاني... الخ.

بعض أدوات FDISK تسمح أيضا بوضع علم التنشيط على الأقسام في الأقراص الثانوية. لكن، تلك الأنظمة لا يمكنها الإقلاع من تلك الأجهزة، بعضها يستخدم القيمة الثابتة التقليدية 80h كعلامة تنشيط، والبعض الأخر يستخدم القيمة المقابلة لوحدة القرص الفيزيائي المعين حاليا 81h, 82h، التي تسمح بالإقلاع من الأجهزة الأخرى، على الأقل، نظريا. في الواقع، هذا سيعمل مع العديد من شفرات MBR، التي تأخذ تعيين بت 7 في خانة حالة الإقلاع كعلم تنشيط بدل القيمة 80h، مع ذلك، سجلات MBR في أنظمة MS-DOS/PC DOS مبرمجة [24] لقبول فقط القيمة الثابتة 80h.

تخزين رقم القرص الفيزيائي الفعلي في جدول الأقسام أيضا سيسبب مشاكل، عندما تتغير إسنادات الأقراص الفيزيائية في BIOS، مثلا عند إزالة، أو إضافة، أو تبديل الأقراص.

ولذلك في MBR العادي قبول بت 7 كعلم تنشيط أو فقط استخدام وتمرير قيمة DL (التي يوفرها في الأصل BIOS) إلى VBR يسمح بقدر كبير من المرونة.

في أنظمة MS-DOS 7.1 – 8.0، سجلات MBR تم تعديلها لتتعامل مع بت 7 كعلم تنشيط، وستعتبر القيم من 01h. إلى 7Fh غير صالحة، لكنها ما زالت تأخذ وحدة القرص الفيزيائي من جدول الأقسام بدل استخدام قيمة DL التي يوفرها نظام BIOS.

سجلات MBR الممتدة (موسعة) في DR-DOS 7.07، تتعامل مع بت 7 كعلم تنشيط وعادة تستخدم وتمرر قيمة BIOS DL (تشمل القيم الغير معيارية 00h..01h التي تستخدمها بعض أنظمة BIOS أيضا من أجل الوسيط المقسم) لكنها توفر أيضا كتلة خاصة لتضبيط محمل الإقلاع NEWLDR لتدعم طرق إقلاع بديلة مع وسيلة LOADER ونظام REAL/32 بالإضافة إلى تغيير سلوك MBR، حتى يستطيع أيضا العمل مع قيم القرص المأخوذة من جدول الأقسام (المطلوبة مع LOADER و AAP، [30] انظر: NEWLDR عند 00Ch+)، وترجمة وحدات قرص Wyse الغير معيارية من 02h..7Fh إلى 80h..FDh، وضبط اختياريا قيمة القرص (المخزنة عند الحيد 19h+ في كتلة EBPB أو عند حيد القطاع 1FDh+) في VBRs المحملة قبل تمرير التنفيذ إليها (انظر: NEWLDR عند الحيد 014h+)، هذا أيضا يسمح لمحملات الإقلاع الأخرى استخدام NEWLDR كمحمل chain-loader، و تضبيط صورتها في الذاكرة أثناء ذلك وتمرير "tunnel" تحميل EBRs، VBRs أو [30] AAPs عبر NEWLDR.

- سجل MBR يجب أن يحفظ محتويات DH و ES:DI لدعم كامل تقنية قبس وتشغيل PnP (أنظر أعلاه). لكن العديد من سجلات MBR لا تفعل ذلك، بما فيها سجلات أنظمة MS-DOS 2.0 - 8.0 / PC DOS 2.0 - 6.3 و ويندوز أن تي XP/2000. (هذا ليس بمستغرب، إذا علمنا أن تلك الإصدارات من نظام دوس كانت قبل صدور مواصفة PnP BIOS، أيضا المعايير والمواثيق السابقة لا تذكر أية متطلبات لحفظ أي تسجيل آخر غير تسجيل DL) بعض سجلات MBR تضبط DH على القيمة 0.

شفرة MBR تمرر معلومات إضافية إلى VBR في عدة تطبيقات:

- DS: SI = تشير إلى مدخلة 16-بايت في جدول أقسام MBR. (الذي تغير مكانه) والمرتبط بسجل VBR النشيط.

نظام PC-MOS 5.1 يعتمد على هذا في الإقلاع إذا لم يجد قسم يحمل علم الإقلاع في جدول الأقسام.

قطاعات إقلاع أنظمة Multiuser DOS و REAL/32 تستخدم هذا مع وسيلة LOADER في تحديد موقع قطاع إقلاع القسم النشيط (أو محمل إقلاع أخر مثل IBMBIO.LDR في مكان ثابت على القرص،) إذا لم تعثر على ملف الإقلاع LOADER.SYS . أنظمة PTS-DOS 6.6 و S/DOS 1.0 تستخدم هذا مع ميزة AAP (القسم النشيط المتقدم)[30].

بالإضافة إلى دعمها LOADER و AAPs، أنظمة DR-DOS 7.07 يمكنها استخدام هذا في تحديد أسلوب نفاذ INT 13h الضروري عند استخدام شفرتها المزدوجة [31] VBR CHS/LBA وستقوم بتحديث حقل علم الحالة / قرص الإقلاع في مدخلة القسم وفقا للقيمة المستخدمة فعليا للتسجيل DL.

محملات إقلاع نظام داروين Darwin (boot1fat32، boot1u، boot1h) تعتمد على هذا المؤشر أيضا ولكنها، بالإضافة لذلك، لا تستخدم التسجيل DS، وتفترض عوض ذلك تعيينه إلى 0000h. [34] وهذا سيسبب مشاكل إذا كان تقديرها غير صحيح.

شفرة MBR في أنظمة OS/2 و MS-DOS 2.0 حتى إصدارة 8.0 و PC DOS 2.0 حتى إصدارة 7.10 و ويندوز أن تي XP/2000، توفر نفس هذه الواجهة أيضا، رغم أنها جميعها لا تستخدمها. شفرة MBR في ويندوز فيستا/7 لم تعد توفر هذا المؤشر DS:SI. في حين أن بعض الامتدادات تعتمد فقط على مدخلة 16-بايت نفسها في جدول الأقسام، نجد امتدادات أخرى قد تتطلب تمثيل جميع مدخلات جدول الأقسام 4 (أو 5).

- DS:BP = اختياريا، تشير إلى مدخلة 16-بايت في جدول أقسام MBR (الذي تغير مكانه) والمرتبط بسجل VBR النشيط.

هذا مطابق للمؤشر في DS:SI. (انظر أعلاه) وتوفره شفرة MBRs في أنظمة PC DOS 2.0-7.10 و MS-DOS 2.0-8.0، وويندوز أن تي XP/2000/Vista/7. لكن لا تدعمه معظم شفرات MBR من الطرف الثالث.

في نظام DR-DOS 7.07،عن طريق MBR ممتد (موسع) يمكن توفير واجهة ممتدة مع وسيلة LOADER، باستخدام التسجيلات التالية:

- AX = توقيع سحري يشير إلى وجود هذا الامتداد NEWLDR (0EDCh)
- DL = وحدة قرص الإقلاع (رقم جهاز الإقلاع) (انظر أعلاه)
- DS:SI = تشير إلى مدخلة 16-بايت المستخدمة في جدول أقسام MBR (انظر أعلاه)
- ES:BX = بداية قطاع الإقلاع أو صورة قطاع NEWLDR (عادة تكون 7C00h)
- CX = محجوزة

LOADER.COM (يعرف أيضا باسم : NEWLDR) : محمل إقلاع متعدد، استخدم في أنظمة دوس مثل: دي آر-دوس DR-DOS، ملتي يوزر دوس Multiuser DOS، من عدة شركات مثل: نوفيل Novell، آي ام أس IMS، كالديرا Caldera، ديجيتال ري سيرش Digital Research (البحوث الرقمية)...وغيرها... ملف LOADER.SYS جزء من تنصيب LOADER.COM.

LOADER.EXE : محمل برنامج تشغيل تلقائي يستخدم اختياريا في عملية بدء تشغيل نظام ويندوز ميلينيوم Windows ME.

IBMBIO.COM : اسم ملف شفرة لتهيئة النظام ومشغلات عتاد مدمجة في عدة أنظمة دوس، الملف جزء من PC DOS و 5.0 DR DOS ونسخ أحدث (باستثناء DR-DOS 7.06). وله نفس وظيفة IO.SYS في MS-DOS، أو DRBIOS.SYS في DR DOS 3.31 حتى إصدار 3.41.

راجع أيضا جداول "تخطيط القطاع" أعلاه.

عند استعمال مخطط تقسيم قرص GPT، اللجنة الفنية الفرعية T13 المسؤولة عن معايير واجهة ATA تقترح شفرة سجل إقلاع رئيسي هجين Hybrid MBR مع المواصفة الرابعة لمحرك الأقراص المحسن EDD-4، هذا الاقتراح يوصي بامتداد آخر إلى الواجهة، باستخدام تسجلات المعالج التالية [36]:

- EAX = 54504721h (أي"GPT!")

  شير إلى أن بنية تسليم أو تحويل Hybrid MBR قد تم تمريرها مع التسجيلان DS:SI عوضا عن مدخلة القسم التقليدي في سجل الإقلاع الرئيسي MBR.
- DL = وحدة قرص الإقلاع (رقم جهاز الإقلاع) (انظر أعلاه)
- DS:SI = يشير إلى بنية تسليم (تحويل) hybrid MBR، المؤلفة من مدخلة افتراضية 16-بايت في جدول أقسام MBR. (مع تعيين جميع بتات باستثناء علم الإقلاع عند الحيد 0h+ ونوع القسم عند الحيد 4h+)، متبوعة ببيانات إضافية. هذا يتوافق جزئيا مع الامتداد القديم DS:SI (أنظر أعلاه)، إذا كانت فقط مدخلة القسم 16-بايت مطلوبة، وليس كامل جدول الأقسام من قبل هذه الامتدادات القديمة. بما أن أنظمة التشغيل القديمة (بما فيها سجلاتهم VBRs) لا تدعم هذا الامتداد ولا هي قادرة على معالجة القطاعات التي تتجاوز حاجز 2 تيرابايت [46]، محمل الإقلاع الهجين الذي يمكن GPT سيكون قادر على محاكاة المدخلة الافتراضية 16-بايت في جدول أقسام MBR إذا كان قسم الإقلاع ضمن منطقة 2 تيرابايت الأولى. [14]
- ES:DI = تشير إلى بنية تفحص تنصيب "$PnP" (انظر أعلاه)

بنية تسليم شفرة إقلاع Hybrid MBR

وصف	اختلافاتها عن بنية التسليم التقليدية MBR	طول بايت	إزاحة بايت	رمز تذكري
تعيين إلى 80h (أي، قسم يقبل الإقلاع)	قيمة ثابتة	1	0	Boot Indicator
تعيين إلى FFFFFFh. يجب أن تتجاهل شفرة إقلاع VBR هذا الحقل		3	1	Starting CHS
تعيين إلى نوع نظام القسم القابل للإقلاع (أي، الذي سيتم تعيينه يملك قسم منصب في تخطيط قرص MBR)	دون تغيير	1	4	OS Type
تعيين إلى FFFFFFh. يجب أن تتجاهل شفرة إقلاع VBR هذا الحقل	قيمة ثابتة	3	5	Ending CHS
تعيين إلى FFFFFFFFh.		4	8	Starting LBA
تعيين إلى FFFFFFFFh.		4	12	Size In LBA
تعيين إلى حجم حقل مدخلة القسم في ترويسة أقسام GPT	حقل جديد	4	16	Size Of Partition Entry
تعيين إلى مدخلة قسم الإقلاع (لكن) في GPT .		حجم مدخلة القسم	20	GPT Partition Entry

مع شفرة الإقلاع التقليدي MBR، هذه البنية تتضمن حقول من مدخلة قسم الإقلاع في MBR. وبما أن قسم GPT قد يقع خارج حدود عنونة 32-بت في LBA، تم إضافة حقول جديدة لنقل كامل المعلومات إلى VBR.

راجع أكثر كتيب "جدول أقسام GUID" وملحق e09127r3 من T13، مع مواصفة UEFI.

تحرير سجل الإقلاع الرئيسي

نظرا لأن MBR متواجد فقط على قرص خام، (أي بدون نظام ملفات/تهيئة)، سوف تحتاج إلى أدوات خاصة لتعامل مع القرص، ورغم إمكانية معالجة MBR مباشرة باستخدام أحد برامج تحرير القرص، هناك أدوات يمكنها كتابة شفرة تعليمات محددة مسبقا إلى MBR على القطاع الأول على القرص. في مايكروسوفت دوس برنامج FDISK منذ الإصدارة 5.0، يستخدم خيار سطر الأوامر MBR / لإعادة كتابة شفرة MBR [37] [21]. لكن برمجية FDISK استبدلت لاحقا بالأداة Diskpart. في ويندوز 2000 /XP استخدمت طرفية الاسترداد لإعادة كتابة شفرة MBR بواسطة الأمر fixmbr. وفي ويندوز فيستا،/7 تستخدم بيئة الاسترداد لإعادة كتابة شفرة MBR بواسطة الأمر BOOTREC /FIXMBR. بعض أدوات الطرف الثالث يمكنها المساعدة أيضا في تحرير جدول الأقسام مباشرة، مثل boot-repair في لينكس، أو MBRWizard [15].

في يونكس/لينكس، عادة تستخدم أداة dd (المتوافقة مع معيار POSIX) في القراءة أوالكتابة إلى أي موقع على جهاز تخزين بما فيها قطاع MBR. أيضا في لينكس، يمكن استخدام برنامج ms-sys في تنصيب MBR الخاص بنظام ويندوز. برامج أخرى في لينكس، مثل محمل الإقلاع GRUB 2 أو LILO تستخدم في إعادة كتابة شفرة MBR، بواسطة الطرفية: grub-install أو lilo -mbr. حاليا النسخة الحديثة من برنامج GRUB 2 تتطلب استخدام الأمر grub-install من طرفية نظام التشغيل أو من طرفية القرص الحي/Live USB. البرنامج التراثي GRUB Legacy (في بعض توزيعات لينكس) يمكنه الكتابة إلى MBR بواسطة الطرفية، باستخدام أوامر setup و embed. إضافة لذلك، هناك عدة برامج قادرة على إنشاء نسخ احتياطية من جدول الأقسام الأولية والأقسام المنطقية [33] (في القسم الممتد.) [32] مثل sfdisk المتوفر على قرص SystemRescueCD.

أمثلة:

برنامج sfdisk (في لينكس) [38].

- عمل نسخة احتياطية من جدول أقسام MBR على القرص : sfdisk -d /dev/hda > hda.out
- استعادة تلك النسخة إلى قطاع إقلاع القرص : sfdisk /dev/hda < hda.out
- نسخ جدول الأقسام من قرص إلى أخر، مفيد في إعداد النسخ المرآوية، لكن انتبه هناك احتمال كبير أن يصبح القرص الثاني غير قابل للقراءة/الإقلاع؛ البرنامج لا يعرض للمستخدم أية رسالة تحذير.
- نسخ جدول الأقسام من قرص إلى قرص: sfdisk -d /dev/sda | sfdisk /dev/sdb

برنامج dd. (في لينكس)

- عمل نسخة احتياطية من MBR على القرص : dd if=/dev/sda of=/path/mbr-backup bs=512 count=1
- استعادة النسخة إلى قطاع إقلاع القرص: dd if=/path/mbr-backup of=/dev/sda bs=512 count=1
- حذف كامل MBR على القرص: dd if=/dev/zero of=/dev/sda bs=512 count=1
- حذف فقط شفرة الإقلاع 446 بايت[44] وترك جدول الأقسام (قد يفيد في حالة إعادة تنصيب نظام تشغيل آخر بالكامل): dd if=/dev/zero of=/dev/sda bs=446 count=1
- قراءة سريعة لمعاملات نسخة سجل الإقلاع الرئيسي: file mbr-backup
- عرض كامل 512 بايت في نسخة سجل الإقلاع الرئيسي: hexdump -Cv mbr-backup

path هو المسار مثل: /home/username/Desktop/mbr-backup و mbr-backup هو اسم الملف (النسخة)

> في حالة تنفيذ الأمر مع جدول أقسام غير متوافق سوف تخسر بيانات القرص وتقريبا يستحيل بعد ذلك استعادتها. لذلك يفضل عوض ذلك، إعادة تنصيب محمل الإقلاع. حذف سجل الإقلاع الرئيسي بالكامل يعني حذف معلومات محمل الإقلاع وجدول الأقسام، مع معلومات أخرى يحتجاها الجهاز في الإقلاع. الجهاز لن يكون قادر على الإقلاع إذا لم يتم مباشرة إعادة تنصيب MBR وشفرة محمل الإقلاع في القطاع الأول.

برنامج ms-sys (في لينكس)

برنامج يمكنه أيضا الكتابة إلى سجلات الإقلاع في أنظمة ويندوز 98/ME/فيستا/7، (بعد تنصيب البرنامج، راجع خياراته ms-sys -h).

- إعادة كتابة معلومات القسم (قطاعات مخفية، رؤوس، وهوية القرص): ms-sys --partition /dev/sda1
- كتابة سجل الإقلاع (على القرص) في أنظمة ويندوز أن تي (دون المس بتوقيع القرص [36]، راجع خيارات كل نظام): ms-sys --mbr /dev/sda
- كتابة قطاع إقلاع جديد (راجع خيارات نوع السجل): ms-sys (1-6)

فيروسات قطاع الإقلاع

رغم أنها كانت مزدهر في بداية التسعينات، [50] فيروسات قطاع الإقلاع اليوم أصبحت نادرة، لأن معظم اللوحات الأم تملك حماية [52] ضد هذا النوع من التهديدات وتمنع ولوج الفيروسات إلى MBR بدون إذن المستخدم. لكن في السنوات الأخيرة، ظهرت برمجيات خبيثة معقدة أوجدت طرق لإبطال تلك الحماية واستهداف مرة أخرى MBR، مثل روتكيت [53]،Rootkit:W32/Whistler.A.

العديد من الفيروسات كانت تكتب خصيصا من أجل سجلات MBR أو VBR، في حالة MBR هذه الفيروسات يمكنها إعادة توجيه، أو إتلاف، أو استبدال [44] MBR بشفرة ضارة، تجعل عملية الإقلاع مستحيلة من القرص الثابت. ولأن شفرة إقلاع MBR يتم تنفيذها بشكل آلي من قبل [20] BIOS، قبل بدء تحميل نظام التشغيل *، هذا الأخير لن يستطيع أبدا اكتشاف أو إصلاح MBR.

(* في معظم الأجهزة، أنظمة التشغيل ليست موجودة في ذاكرة ROM، ولكن تحمل من مكان أخر، مثل القرص الثابت أو الشبكة).

غالبا، فيروسات الإقلاع تنتقل إلى القطاع الأول MBR في القرص الثابت عن طريق البرامج التي تحملها ويتم تفعيلها من داخل النظام أو عند بدء تشغيل الحاسوب عن طريق الأقراص المرنة المصابة، (أو الأقراص الموصولة بمنافذ الناقل التسلسلي العام USB...الخ)، حتى وإن كان ذلك الوسيط غير قابل للإقلاع، يمكنه نقل البرنامج الضار إلى قطاع MBR أو VBR.

نفس المعلومات السابقة تنطبق أيضا على فيروسات VBR التي يتم تنشيطها آليا عند تحديد القسم الأولى النشيط أو تفعيل شفرة إقلاع القسم من قبل MBR قبل تحميل النظام.

إذا حمولة الفيروس لم تعمل أثناء بدأ التشغيل ولم تغير القطاع الأصلي MBR أو VBR، أثناء التشغيل العادي، غالبا نظام التشغيل سيمنع الفيروسات من نسخ نفسها إلى الأقراص الأخرى.

غالبا، هذه الفيروسات المتواجدة في بنية القرص لا يمكنها التأثير على نظام التشغيل لأن النظام لا ينفذ إلى الأقراص الفيزيائية إلا عن طريق مشغلات القرص التي في النمط المحمي.

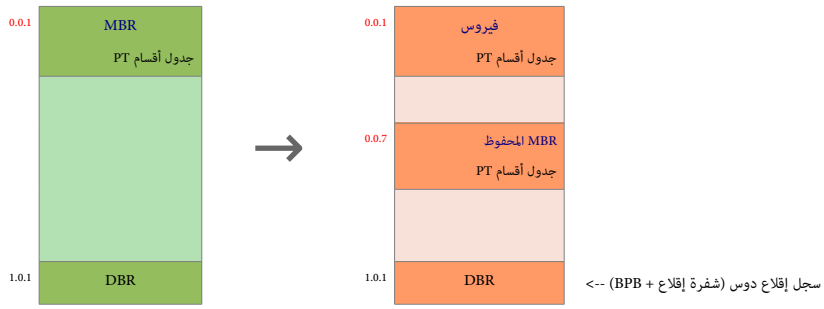
هذه الفيروسات عادة تخرب روتينات النفاذ للقرص BIOS INT 13h، التي يتم تجاهها بعد بدأ تشغيل النظام (مثل ويندوز 2000). لكن في حالة وجود إعدادات لإقلاع عدة أنظمة، مثل وجود ويندوز 2000 إلى جانب م.س.دوس، في هذا الحالة يمكن للفيروس القطاع إصابة الحاسوب عند تشغيل النظام الأخر (مثل، م.س.دوس). إذا حدث ذلك سيكون ويندوز 2000 عرضة للضرر.

الفيروسات التي حمولتها (شفرتها الضارة) تعمل أثناء بدأ التشغيل تشكل خطر على الحاسوب الذي عليه النظام لأنها سوف تنشط قبل أن يسيطر نظام التشغيل على الحاسوب. بعد أن يقوم نظام التشغيل بتفعيل مشغلات القرص التي في النمط المحمي، سوف لن تستطيع الفيروسات نسخ نفسها إلى الأقراص الأخرى أو إلى الأقراص المرنة لأن آلية BIOS التي تعتمد عليها الفيروسات لن تكون مستخدمة في النفاذ إلى القرص.

السيناريو الأول --- شفرة الفيروس في سجل الإقلاع الرئيسي، مع حفظ النسخة الأصلية

فيروسات قطاع الإقلاع غالبا ما تستبدل شفرة إقلاع MBR بشفرتها وتحفظ كامل النسخة الأصلية في مكان آخر على القرص. تلك الفيروسات (مثل فيروس براين وفيروس القدس) بعد تنشيطها تبقى في الذاكرة [51] وتمرر عملية التنفيذ إلى النسخة الأصلية كي يستطيع الجهاز الإقلاع وكي يظهر بدء التشغيل سليم للمستخدم. أكثر الطرق شيوعا في إصابة قطاع الإقلاع، استبدال كامل MBR وحفظ النسخة الأصلية في نهاية القرص الثابت، بعض الفيروسات تخفض حجم القسم كي لا يتم الكتابة مرة أخرى فوق هذه المنطقة. مثلا، فيروس تيكيلا متعدد الأجزاء يتبع هذه التقنية.

مثال آخر: فيروس ستوند الذي يخزن MBR الأصلي في القطاع 7، (أنظر للشكل). بعد أن يحل مكان القطاع الأصلي، يقرأ القطاع 7 في الذاكرة وينقل التحكم إليه. عادة فيروس ستوند يستفيد من القطاعات الشاغرة بعد MBR (على المسار الأول للقرص). لكن هذا ليس مضمون له. إذا لم يحمي الفيروس القطاع المعدل (الحامل للفيروس)، والذي يتضمن MBR، الاستخدام العادي للحاسوب يمكن أن يعيد كتابته، وهذا أيضا قد يعطل وظيفة إعادة التشغيل أو يمنع بدأ تشغيل النظام. وأحيانا يخرب بعض مناطق القرص الثابت أو القرص المرن مما يجعل القرص غير صالح للقراءة.



تخطيط نموذجي للقرص الثابت قبل وبعد إصابة فيروس ستوند Stoned

السيناريو الثاني --- شفرة الفيروس في سجل الإقلاع الرئيسي، لكن بدون حفظ النسخة الأصلية

بعض الفيروسات تستبدل شفرة إقلاع MBR ولا تلمس جدول الأقسام PT، لكنها أيضا لا تحفظ MBR الأصلي في مكان آخر، هذه الفيروسات تحاكي تماما عمل قطاع الإقلاع الأصلي، مثال على ذلك: الفيروس الروسي SVC 6.0، وفيروس أزوسا، المكتشف في يناير/كانون الثاني عام 1991 في أونتاريو- كندا. هذه الفيروسات لا يمكن التخلص منها بالطرق العادية، لأن النسخة الأصلية من MBR ليست محفوظة. عادة، برامج مكافحة الفيروسات تستخدم نسختها المعيارية المضمنة من شفرة MBR في ردها على هذا التهديد

السيناريو الثالث --- الفيروس يتلاعب بمدخلات جدول الأقسام في سجل الإقلاع الرئيسي

جدول الأقسام PT في MBR من الأهداف السهلة للفيروسات، هذه الفيروسات تستطيع تعديل مدخلة القسم النشيط في جدول الأقسام، وتوجيهها إلى قطاع إقلاع شفرة الفيروس. بهذه الطريقة، سجل الإقلاع الرئيسي سوف يحمل قطاع إقلاع الفيروس، ثم بدوره الفيروس يحمل قطاع الإقلاع الأصلي. مثال على ذلك: فيروس ستارشيب.

بعض الفيروسات المخادعة، مثلا فيروسات Ginger family تتلاعب بمدخلات جدول الإقسام بحيث ينشأ عن ذلك ما يشبه تأثير القسم الحلقي [41] [42]. كما يبدو، هذه الفيروسات المخادعة، سوف تجعل نظام م.س.دوس 4.07.0 يدخل في حلقة لا متناهية عند الإقلاع. وسيحتاج المستخدم إلى الإقلاع من قرص مرن؛ م.س.دوس 3.3x، خالي من الفيروسات أو نظام دوس مختلف عن مايكروسوفت دوس، مثل بي سي دوس.

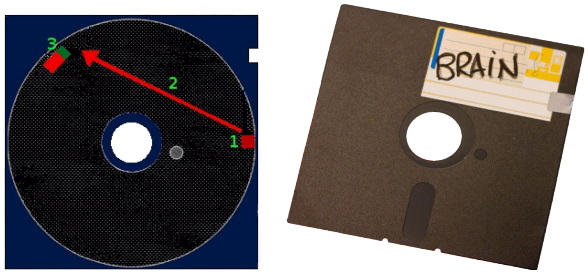
فيروس براين

النوع:	فيروس قطاع إقلاع
المبرمج:	الإخوة باست و أمجد فاروق ألفين
تاريخ الاكتشاف:	يناير/كانون الثاني 1986.01
المكان الأصلي:	لاهور ، باكستان
اللغة الأصلية:	لغة تجميعية
المنصة:	دوس (بي سي دوس أو آي بي إم دوس) نظام ملفات FAT
امتداد العدوي:	من 3,000 إلى 7.000 بايت

فيروس براين (المخ/ الباكستاني، الإنفلونزا الباكستانية، لاهور، فيروس باست) يعتبر أول فيروس على منصة دوس، (وأول فيروس خارج المختبر)، ظهر في يناير/كانون الثاني عام 1986. أول تقرير عن الفيروس كان في 22 أكتوبر/تشرين الأول 1987، عن إصابة أجهزة جامعة ديلوار في الولايات المتحدة، وعدة جامعات في العالم، وحتى شركات. لكن أول ظهور لافت للفيروس كان في ربيع 1988، وفي نوفمبر/تشرين الثاني 1988، أصابة إحدى نسخ الفيروس تقريبا 300 جهاز في جامعة هيوستن. النسخة الأولى من فيروس براين كانت تصيب فقط قطاع إقلاع الأقراص المرنة، مقاس 5.25 بوصة، سعة 360 كيلوبايت (هذا النوع من الأقراص المرنة يميز بواسطة بايت التوقيع عند بداية FAT. الفيروس لا يصيب أنواع الأقراص الأخرى)، عند إدراجها في الحاسوب المصاب (في الذاكرة) [51]. براين كان أيضا أول فيروس متخفي. هذا يعني أنه أول روتكيت [53].

عند إدراج القرص المرن المصاب في الجهاز، ينسخ الفيروس نفسه إلى أعلى منطقة في الذاكرة، (عادة يحتل 7 كيلوبايت) [51]. ولحماية نفسه في RAM، الفيروس سوف يصفر [59]حجم الذاكرة بتبديل متجه المقاطعة 12h [55] [56]. ويصفر متجه المقاطعة 13h للإشارة إلى شفرة الفيروس في أعلى الذاكرة ويصفر كذلك متجه المقاطعة 6Dh [الغير مستخدم في دوس] للإشارة إلى متجه المقاطعة الأصلي 13h. بعدها سوف تستمر عملية الإقلاع المعتادة بتحميل الملفان IBMBIO.COM و lBMDOS.COM في نظام بي سي دوس أو الملفان IO.SYS و MSDOS.SYS في م.س.دوس.

الفيروس (النسخة الأولى) لا يصيب القرص الثابت، لكن أثناء وجوده في الذاكرة، ينقل العدوى إلى الأقراص المرنة المستخدمة. الفيروس يخزن في القطاعات الشاغرة، قطاع الإقلاع الأصلي مع 6 قطاعات ممتدة (3 عناقيد) تحمل متن شفرة الفيروس، ويؤشر عليها بالقطاعات التالفة في نظام FAT. (أنظر للشكل).



قرص مرن، 5.25 بوصة، سعة 360 كيلوبايت

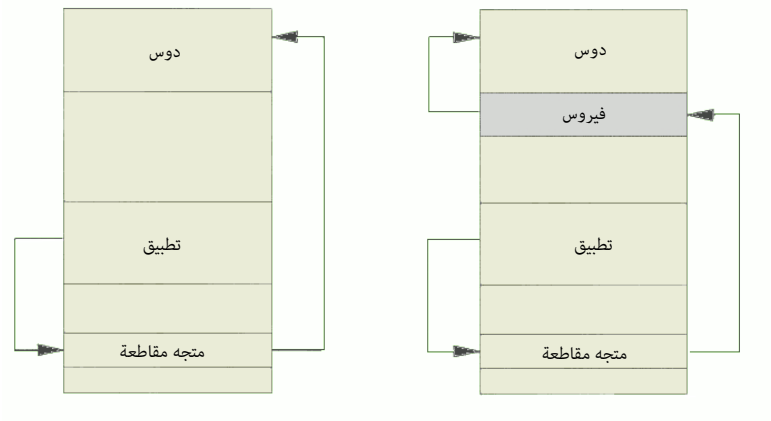
1. سجل إقلاع القرص المصاب VBR
2. قفزة إلى الشفرة الرئيسية
3. سجل إقلاع القرص الأصلي + بقية متن شفرة الفيروس

الأعراض/العدوى:

- تغيير قطاع الإقلاع (الجزء المكشوف من شفرة الفيروس)،
- قطاعات تالفة على القرص المرن، عند العنقود 37h كحد أدنى، وتشكل 3 عناقيد متماسة أو 6 قطاعات.(أول قطاع يتضمن قطاع الإقلاع الأصلي ثم بقية قطاعات الفيروس).
- تمديد في زمن الإقلاع، مع إبطاء عملية النفاذ إلى القرص المرن قد تسبب تعطيلها.
- خسارة النظام 7 كيلوبايت من الذاكرة.
- تغيير في تسمية القرص إلى: "@BRAIN" أو "BRAIN(c)". (هنك نسخة أخرى من الفيروس تغير الاسم إلى "ashar ©" (في شفرة فيروس أشار عند الحيد 04a6h)).

في زمن الإقلاع، فيروس براين يحمل قطاع الإقلاع المصاب مع القطاعات التي يختبأ فيها الفيروس (الموسومة بتالفة!)، الفيروس يحجز 7 كيلوبايت في أعلى منطقة في الذاكرة، بعد أن يحرك ويقرأ قطاع الإقلاع، يقفز إلى شفرة الفيروس.

الفيروس متخفي لأنه يستطيع إعادة توجيه البرامج التي تحاول النفاذ إلى القطاع المصاب إلى قطاع الإقلاع الأصلي المحفوظ، باعتراض أية [55] مقاطعة لقراءة قطاع الإقلاع، عادة بخطف [54] المقاطعة Int 13 من أجل استدعاء شفرة الفيروس، ويعين متجه المقاطعة Int 6Dh إلى متجه المقاطعة الأصلية Int 13 (كي لا يستدعي نفسه عند طلب BIOS). بعد قراءة وتمرير التحكم إلى قطاع الإقلاع الأصلي، يستطيع دوس إتمام عملية الإقلاع. هذا يعني أنك إذا استخدمت برنامج مثل DEBUG، سيبدو لك القطاع سليم. إذا كان الفيروس نشيط في الذاكرة [51]. وأدوات القرص القديمة مثل نورتن لن تستطيع كشفه.



توجيه المقاطعة قبل وبعد حصول الفيروس على التحكم [56]

الفيروس يراقب عمليات Int 13 عند قراءة الأقراص المرنة A أو B. وفقا عملية 'تعداد النفاذ' access count، وقبل أن يصيب الأقراص المرنة، يبحث عن التوقيع 1234h. عند الحيد 04h، المخزن في شكل 12 34، (أنظر للطرح الست عشري) إذا لم يجد التوقيع، ينسخ نفسه إلى القرص بعد أن يحفظ قطاع الإقلاع الأصلي. إذا كنت الوظيفة هي القراءة READ، الفيروس يغير الرأس، والمسار، والقطاع إلى قطاع الإقلاع المحفوظ كي يظل قطاع الإقلاع المصاب مخفي عن أي فحص عرضي. الفيروس لا يستطيع إصابة القرص المحمي من الكتابة.

عادة، الفيروس عند محاولة إصابة القرص، يبحث عن 3 عناقيد متماسة يستطيع وسمها بالتالفة. إذا لم يجد أية عناقيد شاغرة، لا يصيب القرص بالعدوى. لكن إذا وجد عنقود شاغر، وليس أحد العنقودان الأخيران على القرص، يختار ذلك العنقود الشاغر مع العنقودان التاليان له، ثم يوسم الجميع بالتالفة (هذا يعني خسارة البيانات في عنقودين!.).

كما ذكرنا، الفيروس يدس 3 عناقيد تالفة! في منطقة FAT على القرص، وينسخ قطاع الإقلاع الأصلي إلى أول قطاع في القطاعات التالفة ثم ينسخ بقية شفرة الفيروس إلى بقية القطاعات التالفة.

قطاع إقلاع براين، الموجود في الذاكرة، [51] سيحدد موقع العناقيد التالفة في القرص المصاب الجديد. إذا نجح الفيروس في إصابة القرص، تتغير تسمية القرص المرن المصاب إلى Brain @.

لكن بعض أنظمة BIOSs ستعتبر قطاع إقلاع براين غير صالح للإقلاع لأنه لا يملك توقيع 55h AAh في نهاية القطاع (أنظر للطرح أدناه).

قطاع إقلاع القرص المرن المصاب بفيروس براين - القطاع المطلق 0 (الفيزيائي) (CHS 0-0-1)

```
     00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F  0123456789ABCDEF
0000 fa e9 4a 01 34 12 00 05 08 00 01 00 00 00 00 20 |..J.4.......... |
0010 20 20 20 20 20 20 57 65 6c 63 6f 6d 65 20 74 6f |      Welcome to|
0020 20 74 68 65 20 44 75 6e 67 65 6f 6e 20 20 20 20 | the Dungeon    |
0030 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 |                |
0040 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 |                |
0050 20 28 63 29 20 31 39 38 36 20 42 61 73 69 74 20 | (C) 1986 Basit |
0060 26 20 41 6d 6a 61 64 20 28 70 76 74 29 20 4c 74 |& Amjad (pvt) Lt|
0070 64 2e 20 20 20 20 20 20 20 20 20 20 20 20 20 20 |d.              |
0080 20 42 52 41 49 4e 20 43 4f 4d 50 55 54 45 52 20 | BRAIN COMPUTER |
0090 53 45 52 56 49 43 45 53 2e 2e 37 33 30 20 4e 49 |SERVICES..730 NI|
00a0 5a 41 4d 20 42 4c 4f 43 4b 20 41 4c 4c 41 4d 41 |ZAM BLOCK ALLAMA|
00b0 20 49 51 42 41 4c 20 54 4f 57 4e 20 20 20 20 20 | IQBAL TOWN     |
00c0 20 20 20 20 20 20 20 20 20 20 20 20 4c 41 48 4f 52|            LAHOR|
00d0 45 2d 50 41 4b 49 53 54 41 4e 2e 2e 50 48 4f 4e |E-PAKISTAN..PHON|
00e0 45 20 3a 34 33 30 37 39 31 2c 34 34 33 32 34 38 |E :430791,443248|
00f0 2c 32 38 30 35 33 30 2e 20 20 20 20 20 20 20 20 |,280530.        |
0100 20 20 42 65 77 61 72 65 20 6f 66 20 74 68 69 73 |  Beware of this|
0110 20 56 49 52 55 53 2e 2e 2e 2e 43 6f 6e 74 61 |  VIRUS.....Conta|
0120 63 74 20 75 73 20 66 6f 72 20 76 61 63 63 69 6e |ct us for vaccin|
0130 61 74 69 6f 6e 2e 2e 2e 2e 2e 2e 2e 2e 2e 2e 2e |ation...........|
0140 2e 2e 2e 2e 20 24 23 40 25 24 40 21 21 20 8c c8 |.... $#@%$@!! ..|
0150 8e d8 8e d0 bc 00 f0 fb a0 06 7c a2 09 7c 8b 0e |..........|..|..|
0160 07 7c 89 0e 0a 7c e8 57 00 b9 05 00 bb 00 7e e8 |.|...|.W......~.|
0170 2a 00 e8 4b 00 81 c3 00 02 e2 f4 a1 13 04 2d 07 |*..K..........-.|
0180 00 a3 13 04 b1 06 d3 e0 8e c0 be 00 7c bf 00 00 |............|...|
0190 b9 04 10 fc f3 a4 06 b8 00 02 50 cb 51 53 b9 04 |..........P.QS..|
01a0 00 51 8a 36 09 7c b2 00 8b 0e 0a 7c b8 01 02 cd |.Q.6.|.....|....|
01b0 13 73 09 b4 00 cd 13 59 e2 e7 cd 18 59 5b 59 c3 |.s.....Y....Y[Y.|
01c0 a0 0a 7c fe c0 a2 0a 7c 3c 0a 75 1a c6 06 0a 7c |..|....|<.u....||
01d0 01 a0 09 7c fe c0 a2 09 7c 3c 02 75 09 c6 06 09 |...|....|<.u....|
01e0 7c 00 fe 06 0b 7c c3 00 00 00 00 32 e3 23 4d 59 ||....|.....2.#MY|
01f0 f4 a1 82 bc c3 12 00 7e 12 cd 21 a2 3c 5f 0c 05 |.......~..!.<_..|
     00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F  0123456789ABCDEF
```

براين، أول وأحد الفيروسات النادرة الذي تضمن اسم، ورقم هاتف، وعنوان المبرمج الحقيقي؛ الإخوة باسط وأمجد فاروق ألفين من لاهور، باكستان. ألذان كتب الفيروس لإصابة الأجهزة التي تستخدم النسخ المقرصنة من برنامجهما. فيروس ميرفي من بلغاريا تضمن أيضا رقم هاتف وعنوان الكاتب في الشفرة الأصلية.

كنية فيروس براين في برامج المضادة للفيروسات

Avast!: Brain	Doctor Web: Brain.dropper	Grisoft: Brain	Panda: Brain.1986	Sophos: Brain drop
Avira: Brain #2	F-Prot: BOOT SECTOR DROPPER	Kaspersky Lab: Virus.Boot.Brain.a or Brain.a	RAV: Brain.A	Symantec: Brain
ClamAV: Brain.2	F-Secure: Brain	McAfee: BtDr.Brain	Bitdefender: Trojan.Dropper.Boot.Brain.A	Trend Micro: (C)BRAIN

تنويعات (نسخ) فيروس براين Brain

هناك أشكال مختلف من هذا الفيروس ظهرت لكن أغلبها غير ضار. أحدها كان مصمم كي ينشط في 5 مايو/ أيار 1992. وكان بحجم أكبر ومعظم شفرة الفيروس كانت تقع في القطاعات التي تحمل علامة قطاع تالف في نظام ملفات FAT. وكان قطاع الإقلاع الأصلي يخزن أيضا في تلك القطاعات.

- فيروس براين بي، فيروس هيوستن، أو فيروس القرص الثابت: يصيب القرص الثابت.
- فيروس براين سي: نسخة من براين بي بدون التسمية Brain © (يصيب القرص الثابت، لكن لا يغير تسمية القرص).
- فيروس براين مستنسخ : يشبه براين سي. لكن تم استبدال رسالة الفيروس بشفرة لا يمكن طباعتها تبدو مثل المحارف العشوائية.
- فيروس براين مستنسخ بي: نسخة من براين مستنسخ. تخرب نظام FAT إذا تم الإقلاع بعد تاريخ 1992.05.05.
- فيروس براين شو: هذه النسخة تشبه براين بي، باستثناء الرسالة المعدلة (أنظر أسفل).

```
Welcome to the Dungeon
© 1986  Brain & Amjads (pvt) Ltd.
VIRUS_SHOE RECORD v9.0
Dedicated to the dynamic memories
of millions of virus who are no longer with us today -
Thanks GOODNESS!!
BEWARE OF THE er..VIRUS :This program is catching
program follows after these messeges.....
$# @%$@!!
```

- فيروس أشار، بعض المصادر تقول أنها قد تكون أقدم من النسخة الأصلية.
- فيروس براين شو بي: لا يصيب القرص الثابت، ونسخة أخرى من هذه تحمل الرقم v9.1 بدل v9.0. (والخبراء مختلفون في تصنيف هذا الفيروس.).
- فيروس براين شو وحيد: نسخة من فيروس براين شو في هذه التنويعة، اقتطعت الرسالة إلى سطر واحد
- فيروس براين جورك نسخة من براين شو مع النص: "Welcome to the Dungeon (c) 1986 Brain" عند الحيد 0010h و : "(c) 1986 Jork & Amjads (pvt) Ltd" عند 0202h
- فيروس براين سنغافورة: في هذه، تاريخ حقوق التأليف والنشر يعود إلى عام 1988 وليس عام 1986. مع نفس معلومات الكاتب ولكن النص مختلف:

```
Ver ـ(Singapore) Beware of this "virus". It will transfer to a million of Diskettes... $# @%$@!!
```

فيروس ستوند

النوع:	فيروس قطاع إقلاع
المبرمج:	؟
تاريخ الاكتشاف:	1988.02
المكان الأصلي:	ولينغتون، في نيوزيلندا
اللغة الأصلية:	لغة التجميع
المنصة:	دوس
امتداد العدوي:	512 بايت

فيروس ستوند اسم لعائلة كبيرة من فيروسات قطاع الإقلاع يعود تاريخها إلى بداية عام 1988. كان أبرزها فيروس مايكل أنجلو السيئ السمعة، الذي أحدث ضجة كبيرة في أوائل التسعينات 1990s. وفيروس أنجلينا من عام 1994 الذي عاود الظهور مرة أخرى في عام 2007 في أجهزة الحاسب المحمول.

القطاع المطلق 0 (الفيزيائي) (CHS 0-0-1)

```
     00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F 0123456789ABCDEF
0000 EA 05 00 C0 07 E9 99 00 00 51 02 00 C8 E4 00 80 ê..À.é™..Q..Èä.€
0010 9F 00 7C 00 00 1E 50 80 FC 02 72 17 80 FC 04 73 Ÿ.|...P€ü.r.€ü.s
0020 12 0A D2 75 0E 33 C0 8E D8 A0 3F 04 A8 01 75 03 ..Òu.3ÀŽØ ?.¨.u.
0030 E8 07 00 58 1F 2E FF 2E 09 00 53 D1 52 06 56 57 è..X..ÿ...SÑR.VW
0040 BE 04 00 B8 01 02 0E 07 BB 00 02 33 C9 8B D1 41 ¾..¸....»..3É‹ÑA
0050 9C 2E FF 1E 09 00 73 0E 33 C0 9C 2E FF 1E 09 00 œ.ÿ...s.3Àœ.ÿ...
0060 4E 75 E0 EB 35 90 33 F6 BF 00 02 FC 0E 1F AD 3B Nuàë5.3ö¿..ü...;
0070 05 75 06 AD 3B 45 02 74 21 B8 01 03 BB 00 02 B1 .u..;E.t!¸..»..±
0080 03 B6 01 9C 2E FF 1E 09 00 72 0F B8 01 03 33 DB .¶.œ.ÿ...r.¸..3Û
0090 B1 01 33 D2 9C 2E FF 1E 09 00 5F 5E 07 5A 59 5B ±.3Òœ.ÿ..._^.ZY[
00A0 C3 33 C0 8E D8 FA 8E D0 BC 00 7C FB A1 4C 00 A3 Ã3ÀŽØúŽÐ¼.|û¡L.£
00B0 09 7C A1 4E 00 A3 0B 7C A1 13 04 48 48 A3 13 04 .|¡N.£.|¡..HH£..
00C0 B1 06 D3 E0 8E C0 A3 0F 7C B8 15 00 A3 4C 00 8C ±.ÓàŽÀ£.|¸..£L.Œ
00D0 06 4E 00 B9 B8 01 0E 1F 33 F6 8B FE FC F3 A4 2E .N.¹¸...3ö‹þüó¤.
00E0 FF 2E 0D 00 B8 00 00 CD 13 33 C0 8E C0 B8 01 02 ÿ...¸..Í.3ÀŽÀ¸..
00F0 BB 00 7C 2E 80 3E 08 00 00 74 0B B9 07 00 BA 80 ».|.€>...t.¹..º€
0100 00 CD 13 EB 49 90 B9 03 00 BA 00 01 CD 13 72 3E .Í.ëI.¹..º..Í.r>
0110 26 F6 06 6C 04 07 75 12 BE 89 01 0E 1F AC 0A C0 &ö.l..u.¾‰...¬.À
0120 74 08 B4 0E B7 00 CD 10 EB F3 0E 07 B8 01 02 BB t.´.·.Í.ëó..¸..»
0130 00 02 B1 01 BA 80 00 CD 13 72 13 0E 1F BE 00 02 ..±.º€.Í.r...¾..
0140 BF 00 00 AD 3B 05 75 11 AD 3B 45 02 75 0B 2E C6 ¿...;.u.;E.u..Æ
0150 06 08 00 00 2E FF 2E 11 00 2E C6 06 08 00 02 B8 .....ÿ....Æ.....¸
0160 01 03 BB 00 02 B9 07 00 BA 80 00 CD 13 72 DF 0E ..»..¹..º€.Í.rß.
0170 1F 0E 07 BE BE 03 BF BE 01 B9 42 02 F3 A4 B8 01 ...¾¾.¿¾.¹B.ó¤¸.
0180 03 33 DB FE C1 CD 13 EB C5 07 59 6F 75 72 20 50 .3ÛþÁÍ.ëÅ.Your P
0190 43 20 69 73 20 6E 6F 77 20 53 74 6F 6E 65 64 21 C is now Stoned!
01A0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
01B0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
01C0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
01D0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
01E0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
01F0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
     00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F 0123456789ABCDEF
```

512 بايت، من فيروس ستوند آي Stoned.A (في القرص المرن، مع حيز من أجل جدول الأقسام، أي يمكن أن يأثر على MBR على القرص الثابت).

إذا كان إقلاع الحاسوب من قرص غير القرص الثابت، الفيروس ينتقل إلى الذاكرة، [51] ويتفحص سجل الإقلاع الرئيسي على القرص الثابت، إذا وجده سليم يصبه. عند إصابة القرص المرن، يحرك VBR إلى القطاع 11 ثم ينقل نفسه إلى القطاع 0. عند إصابة القرص الثابت، يحرك MBR إلى القطاع 0/0/7 ويضع نفسه في القطاع 0/0/1. فيروس ستوند يصيب فقط الأقراص المرنة، سعة 360 كيلوبايت، مقياس 5.25 بوصة، والأقراص الثابتة. إذا كان الفيروس في الذاكرة [51]. سوف يصيب قطاع إقلاع أي قرص مرن يستخدم. إذا حذف الفيروس من MBR، لن يعاود إصابة القرص الثابت، حتى وإن كان موجود في الذاكرة.

هناك احتمال أن ينقل الفيروس حمولته عند الإقلاع، حينها الحاسوب المصاب سيصدر صوت صفير ويعرض هذه الرسالة: Your PC is now stoned! LEGALIZE MARIJUANA!

الفيروس لا يسبب أي ضرر، لكن عند نقله النسخة الأصلية من قطاع الإقلاع إلى القطاع 11 على القرص المرن، 5.25 بوصة، المستخدم سيخسر الملفات الموجودة هناك. بعض نسخ دوس تستخدم القطاع 11 كجزء من نظام FAT، وهذا سيخرب FAT على القرص.

يعتقد أن فيروس ستوند من تصميم طالب في جامعة ولينغتون، في نيوزيلندا، وأن تاريخ وجوده يعود إلى عام 1987، لكن أقدم تاريخ مدون في شفرة الفيروس هو فبراير/شباط 1988.

هناك من يدعي أن وزارة الدفاع نيوزيلندية كانت المسؤولة عن انتشار الفيروس في أستراليا. لأسباب سياسية. لكن وزارة الدفاع نيوزيلندية نفت التهمة أنذاك.

تنويعات (نسخ) فيروس ستوند

أشكال مختلفة ظهرت من هذا الفيروس المشهور، أبرزها فيروس مايكل أنجلو السيئ السمعة، الجدول التالي يظهر فقط أهم التنويعات الأخرى لهذا الفيروس:

Angelina (MBR/VBR)	Stoned.C (MBR)	Stoned.D (MBR)	Alive (MBR)	AntiExe (MBR)	Antigame (MBR)	Azusa (MBR/VBR)	Bloomingtion (MBR)	Beijing (MBR/VBR)	Copy77 (MBR/VBR)
Daniela (MBR/VBR)	Dinamo (MBR)	DiskWash (MBR)	Ebpr (MBR)	EmpireMonkey (MBR/VBR)	Face (VBR)	Flame (MBR)	GKCHP (MBR)	Hysteria (MBR)	IntFF (MBR)
Lch15 (MBR)	LovChild	Love (A/B) (MBR)	Manitoba (MBR/VBR)	May21 (VBR)	Million (MBR/VBR)	NearDark (MBR)	Nov7 (MBR)	PC-AT (MBR)	Rostov (MBR/VBR)
Scale (MBR)	Sex (VBR)	Spook (MBR)	Swedish (MBR)	Torm (MBR/VBR)	WXYC (VBR)	Zappa (MBR)	Zapped (MBR)		

كشف وإصلاح القطاع

يمكنك استخدام أية أداة لاستكشاف القرص وعرض شفرة MBR، ومقارنتها بنسخة من قطاع إقلاع سليم . هناك أدوات أخرى في ويندوز تستخدم لحفظ واستعادة MBR. سنذكر بعضها لاحقا.

أدوات عدة يمكن استخدمها لإصلاح MBR المتضرر على القرص، كي تستطيع الولوج إلى وحدات التخزين/النظام. اختيار الأداة المناسبة يعتمد على نوع الضرر الحاصل في قطاع إقلاع MBR / VBR، وما إذا كان جدول الأقسام متضرر أيضا، أو بدأ تشغيل يعمل في نظام مثل ويندوز 2000.

إذا لم يقلع الحاسوب، يجب أولا تشغيل برنامج لاستعراض القرص (مثل dd مع hexdump)، عن طريق أي وسيط صالح، وتفحص القطاع الفيزيائي الأول:

- تأكد أن MBR ليس معبأ بالأصفار أو معبأ بأي محرف أحادي آخر.
- تأكد من وجود رسائل الأخطاء (مثل "Invalid partition table") في شفرة أسكي.
- تأكد من وجود توقيع 0x55AA في نهاية قطاع (512 بايت).

أسهل طريقة لإصلاح أو إعادة كتابة MBR هي استخدام برنامج مايكروسوفت [25] FDISK مع الخيار MBR /، كما في الخطوة التالية من القرص المرن: A:\> FDISK.EXE /MBR #

البرنامج المعياري FDISK موجود في أنظمة مايكروسوفت دوس، و ويندوز ME/95/98. لكن قبل تنفيذ الأمر fdisk /mbr يجب أن تعلم:

- الفيروسات قد تكون في الملفات العادية وفي MBR أو قطاع الإقلاع، واستعادة MBR في هذه الحالة لا يحل المشكلة إذا عاودت الفيروسات فورا الانتشار في النظام.
- الأقراص الديناميكية [28] أو أقراص GPT لا تدعم تنفيذ هذا الأمر.
- تنفيذ الأمر في مايكروسوفت دوس سيعيد كتابة فقط 446 بايت الأولى من MBR، المعروفة باسم شفرة الإقلاع [44]، وسيبقى جدول الأقسام دون تغيير (دون إصلاح).

في حالة نظام مثل ويندوز أن تي XP/2000، يمكنك الإقلاع عن طريق القرص المرن أو القرص المدمج، وتحديد خيار الإصلاح، ثم من طرفية استعادة النظام، نفذ أمر FIXMBR. يمكن أيضا استخدام برامج من طرف ثالث، أو إذا كانت تملك نسخة احتياطية من MBR، يمكنك نسخها إلى القطاع الأول باستخدام إحدى الإداوات أو قرص لينكس الحي، (راجع أدوات dd و hexdump).

> طرفية الاسترداد، وسيلة لحل المشاكل في نظام ويندوز، هذه الطرفية توفر الأمر fixmbr، الذي يعمل تماما مثل الأمر fdisk /mbr ، ويستبدل فقط شفرة الإقلاع ولا يمكنه التأثير على جدول الأقسام. لهذا السبب، لا يمكنه المساعدة في حل مشكلة فيروس سجل الإقلاع الرئيسي. لمعلومات أكثر عن هذه الأدوات وغيرها راجع موقع دعم مايكروسوفت الرسمي.

استعادة MBR عن طريق DiskProbe

استعادة MBR مع جدول الأقسام باستخدام أداة DiskProbe ممكن لكن بشرط توفر نسخة احتياطية للقطاع (512 بايت)، مع إمكانية تشغيل نظام ويندوز.

في حالة توفر نسخة احتياطية سابقة من MBR عن طريق DiskProbe، يمكنك استخدامها مرة أخرى لاستعادة MBR على أي قرص لا يستخدم في بدأ تشغيل الحاسوب. استعادة هذه النسخة سيعيد كتابة كامل القطاع، بما في ذلك جدول الأقسام. هذه الأداة تعمل فقط في أنظمة XP/ 2000 و NT 4.0. ولا تعمل في مايكروسوفت دوس، أو ويندوز ME/95/98. لا يمكن استخدام هذه الأداة، إذا كان تلف MBR على القرص الذي عليه ويندوز، ولا يمكن بدأ التشغيل. في هذه الحالة، يجب استخدام طرفية الاسترداد في إصلاحه.

استبدال جدول الأقسام عن طريق محرر للقرص من طرف ثالث

لإصلاح بدأ تشغيل ويندوز، يمكن استخدام محرر للقرص على مستوى منخفض يرتكز على مايكروسوفت دوس. هذا الأسلوب يحتاج إلى خبرة في تحرير جدول الأقسام يدويا. حتى تستطيع إصلاح جدول الأقسام، يجب معرفة القيم الصحيحة التي ستستخدم في إعادة إنشاء جدول الأقسام. إذا كانت هناك نسخة احتياطية من MBR وجدول الأقسام مصدرها DiskProbe، وكانت تلك النسخة على قرص مرن أو على حاسوب آخر، حينذاك يمكنك استخدام DiskProbe على حاسوب أخر لمشاهدة القيم الصحيحة بحيث تستطيع يداويا إعادة إنشاء جدول الأقسام.

لا تنسى أن تلف القطاع الأول على القرص قد يكون سببه خطأ من المستخدم، أو مشكلة في العتاد (أي الخردة الصينية)، أو مشكلة في التيار الكهربائي، بالإضافة إلى الفيروسات، وعوامل أخرى.

أخيرا، المعلومات الوارد أعلاه أغلبها مرتبط بأنظمة ويندوز ودوس القديمة. ومشاكل الإقلاع غالبا ما يتم إصلاحها عن طريق استخدام قرص التنصيب أو تنفيذ الأمر مثل: chkdsk c: /r..

راجع أيضا فقرة: تحرير سجل الإقلاع الرئيسي (أعلاه).

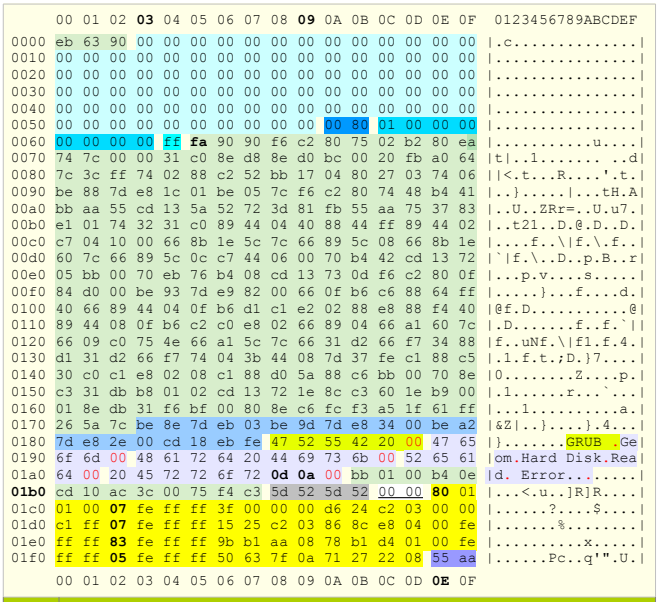
> عمل نسخة احتياطية من MBR وتخزينها على وسيط آخر لاستعادتها لاحقا، قد يفيد في حالة كتابة القطاع بالخطأ، أو تلفه بفيروس. لكن إذا أردت النصيحة، لا تستخدم أداوت قديمة أو برامج ليست مصممة للنظام، أو برامج طرف ثالث مصدرها غير موثوق (مثل موقع إنترنت)

سجل الإقلاع في توزيعات لينكس: (برنامج GRUB 2)

تنصيب محمل الإقلاع GRUB 2 في MBR لا يعني تنصيب كامل البرنامج في القطاع الأول. لأن حجم القطاع 512 بايت فقط والبرنامج أكبر من ذلك. لهذا السبب غالبا ما تستخدم 62 قطاع الشاغرة التي تأتي مباشرة بعد القطاع الأول على المسار الأول من القرص (علما أن عدد القطاعات الشاغرة عند بداية القرص ارتفع في برامج التقسيم الحديثة [47]). تنصيب أو إعادة تنصيب محمل الإقلاع في MBR يعني نسخ صورة boot.img إلى منطقة 446 بايت داخل MBR، وتضمين صورة core.img في 49 قطاع تقريبا التي تلي في المسار الأول. لكن أحيانا قد لا يرغب المستخدم في تضمين core.img في المسار الأول من القرص، لأن تلك المنطقة تشغلها شفرة مثل وحدات RAID أو إحدى برمجيات تشفير كامل القرص. في مثل هذه الحالات. يبحث المستخدم عن خيارات أخرى، عند تنصيب توزيعة لينكس بعد تنصيب ويندوز، سوف تستبدل شفرة ويندوز [44] في MBR بشفرة محمل الإقلاع، لكن يحافظ على البنية التقليدية للقطاع التي سوف تشبه البنية التالي:

- تعليمة القفزة JMP، إلى متن شفرة تنفيذية. (هذه ستكون جزء من شفرة محمل الإقلاع)
- حيز خاص بمعاملات BPB. (غير ضرورية في MBR؛ لكن برنامج grub-install يحجز لها مكان في الشفرة لأنه يستخدم نفس صورة VBR، بالإضافة لاستغلالها في معاملات DAP)
- بعض القيم الأولية المهمة في محمل الإقلاع. (ستكون بحجم 12 بايت في الشفرة التراثية GRUB Legacy و 11 بايت في شفرة GRUB 2)
- متن شفرة محمل الإقلاع الرئيسية. وتشمل أيضا:
  - شفرة لمعالجة عرض رسائل الأخطاء.
  - سلسلة محارف هوية: GRUB + رسائل الأخطاء الموجزة.
  - الروتين الثانوي لعرض المحارف.
- توقيع للقرص (هذا الرقم التسلسلي للقرص يستخدم أصلا في مايكروسوفت أن تي وأنظمة أخرى!) [36].
- جدول الأقسام (على القرص المرن) -- في القرص المرن ستكون بقية شفرة الإقلاع.
- توقيع الاقلاع.

3 بايت EB 63 90 (تعليمة القفزة) في بداية الطرح الست عشري/أسكي التالي، تدل على وجود شفرة محمل الإقلاع GRUB 2 في MBR، هذه 446 بايت تتضمن أيضا حيز لكتلة معاملات BPB (في الأصل، تستخدمها، أنظمة ويندوز في VBR) من الحيد 0x03h إلى 0x5a، ثم عنوان نواة محمل الإقلاع في الذاكرة (2 بايت) حيث يتم نسخ مرحلة الإقلاع التالية من القرص الثابت لتنفيذها. ثم عنوان قطاع مرحلة الإقلاع التالية على القرص الثابت (8 بايت) ثم جهاز الإقلاع الذي تحمل منه النواة (1 بايت). ثم بقية مكونات الشفرة الإقلاع التي تستمر حتى بداية جدول الأقسام (64 بايت) عند الحيد **0x1be**. (سجل MBR التقليدي يدعم فقط 4 مدخلات، كل مدخلة بحجم 16 بايت). الطرح التالي مثال على شفرة GRUB 1.98 في MBR:



بايت	شفرة أسكي (محرف) / بايت ست عشري			
1	CRLF	Carriage Return	**0D**h (13)	محرف رجوع إلى السطر، مرجع إلى السطر
1		Line Feed	**0A**h (10)	محرف تزويد سطر
1	Null character (sz)		00h (00)	سلسلة نهاية صفرية [60]

لمعلومات أكثر عن بنية MBR في لينكس راجع موقع الدليل الرسمي للبرنامج GRUB 2 والشفرة في موقع الأرشيف.

ملاحظات

1. ^ أ، ب، ت، ث، ج، ح، خ، في قطاعات الإقلاع، توقيع الإقلاع أو الرقم السحري 0xAA55، (ويسمى أيضا: توقيع MBR أو توقيع القطاع في MBR/EBR/VBR. أو VP في OS/2) سيكون دائما عند الحيد 1FEh (55h) عند 1FEh و AAh عند 1FFh). هذا التوقيع استخدم أول مرة في الأقراص المرنة في نظام دوس 2.00 عام 1983، ويشير إلى صحة شفرة الاقلاع المتوافقة مع أنظمة اي بي ام،

تقنيا،هذا التوقيع مجرد مؤشر أو معرف، يمكن معاملته كـ 2 بايت منفصلة، (كما في سطر شفرة التجميع (db 55h, AAh) أو ككلمة واحدة 0xAA55 أو 0AA55h. لكن انتبه قيمة AA55h ـ(43605) ليست كالقيمة 55AAh ـ(21930). لذلك ترتيبها مهم جدا.	إزاحة (ضمن القطاع)		قيم البايت (نظام ست عشري)	شفرة أسكي (محرف)
	عشري	ست عشري		
	510	1FE	55	U
	511	1FF	AA	لا يوجد محرف معياري في شفرة أسكي للرقم 0xaa

في إضافات مثل بطاقة العرض المرئي ما زال يستخدم نفس المعرف 55 AA، في شفرة BIOS المحملة لكن الموقع في بداية الشفرة وليس نهايتها كما يظهر في هذا الطرح:

جزء من طرح VGA BIOS

```
      00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F  0123456789ABCDEF
0C0000 55 AA 40 EB 04 37 34 30 30 E9 0A 15 00 00 00 00 |U.@..7400.......|
0C0010 00 00 00 00 00 00 00 00 00 00 00 00 00 00 49 42 |..............IB|
0C0020 4D 20 56 47 41 20 43 6F 6D 70 61 74 69 62 6C 65 |M VGA Compatible|
0C0030 20 42 49 4F 53 2E 20 00 BB 66 DB 01 EF 01 81 00 | BIOS. ..f......|
0C0040 . . .
```

بطاقة VGA تشتغل بواسطة برنامج يدعى VGA BIOS (امتداد من نظام BIOS) الذي يحمله في عنوان خاص في ذاكرة النظام من برنامج ثابت Option ROM[62] يقع على بطاقة VGA.

معظم محملات الاقلاع الموجودة في MBR أو BIOS تتفحص هذا التوقيع قبل تمرير التنفيذ إلى شفرة الاقلاع في قطاع الاقلاع (لكن هناك أنظمة لا تفعل ذلك، مثل IBM PC ROM-BIOS) ومهما كانت قيمة عدد بايتات كل قطاع. هذا التوقيع سيظل دائما عند الحيد 1FEh (أي عند بايتات 510-511). إذا كان القطاع بحجم 512 بايت فهذا يعني أيضا نهاية القطاع.

العديد من وثائق FAT تذكر أن التوقيع 0xAA55 يحتل 2 بايت الأخيرة من القطاع، لكن هذه المعلومة صحيحة فقط إذا كان القطاع بحجم 512 بايت. وإذا كان أكبر من 512 بايت، لن تتغير إزاحة التوقيع (أيضا لا مشكلة إذا تضمنت مرة أخرى 2 بايت الأخيرة من قطاع الاقلاع الأكبر هذا التوقيع).

```
01E0 . . .
01F0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 55 AA [..............U.]
     00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F  0123456789ABCDEF
```

بما أن تمثيل ترتيب البيانات يجب أن يكون نهوي صغير في الأجهزة المتوافقة مع أنظمة IBM، توقيع (16-بت) سوف يكتب بالشكل AA55h في برامج x86 (لحظ ترتيبها المعكوس)، بينما سيكتب بالشكل 55AAh في برامج المعالجات الأخرى التي تستخدم نهوي كبير. ونظرا لأن هذه الطرق في التمثيل قد تم الخلط بينها في الكتب والمراجع الأصلية من مايكروسوفت. المعلومات الواردة هنا من "الموسوعة الحرة" سوف تستخدم طريقة البايت المرتكز على الإزاحة في تمثيل البيانات على القرص لتجنب أية تفسير خاطئ.

للتوضيح أكثر، هذه أمثلة لتطبيق التوقيع على منصات مختلفة مع طرق نهوية مختلفة:

- التوقيع الأخير في القطاع الثاني في سجل إقلاع FAT32 يستخدم توقيع بقيمة 4 بايت: AA550000h حتى يعتبر مضمون القطاع صالح. 2 بايت العليا في هذه القيمة تحتل نفس موقع إزاحة البايت 510 و 511 عند 1FEh المستخدمة في توقيع القطاع 0.
- MBR في جهاز أتاري ويدعى قطاع الجذر Root Sector؛ للدلالة على قابلية قطاع الاقلاع على الإقلاع، يستخدم تدقيق المجموع (16-بت) الذي يجب أن يساوي القيمة $1234. (4660) في 2 بايت الأخيرة في القطاع بدلا من التوقيع 0x55 0xAA المستخدم في الحاسوب الشخصي. خوارزمية تدقيق المجموع تعالج كلمة 256 (نهوي-كبير) في قطاع الاقلاع 512 بايت وتشمل 2 بايت الأخيرة، إذا كانت النتيجة الرقم السحري $1234 يصبح القطاع قابل للإقلاع. (معالجات موتورولا 680x0 المستخدمة في أجهزة أتاري تستخدم طريقة نهوي-كبير في تمثيل البيانات في الذكرة ويجب أخذ هذا التمثيل بالاعتبار عند حساب تدقيق المجموع).

لكن ماذا عن هذه النهوية؟ دون الخوض في التفاصيل، النهوية هي طريقة ترتيب البيانات داخل ذاكرة الحاسوب أو طريقة تخزينها على الوسيط (مثل القرص).

تذكر أن الأرقام (القيم الست عشري) الممثلة في أكثر من بايت واحد تخزن/تظهر غالبا بطريقة نهوي صغير أو بترتيب ثمانيات معكوس (البايت يرتكز على الإزاحة)، لكن في بعض الأمثلة الأخرى تظهر بطريقة نهوي كبير، وكما ذكرنا أعلاه، هذا الخلط هنا ليس مقصود. وسببه اختلف تلك المصادر. في أنظمة إنتل x86 أرقام 2-بايت تخزن في الذاكرة بحيث يكون البايت الأدنى أولا LSB [58] و البايت الأعلى آخرا MSB [58]. كما في مثال، توقيع قطاع الاقلاع عند الحيد 1FEh، مثال آخر، هذان الملفان (من أداة سطر الأوامر لمراقب الشبكة Tcpdump) تعرض الاختلاف بين تمثيل نهوي كبير ونهوي صغير مع نفس البيانات في نظام يونكس من إنتل ونظام سولاريس من سن ميكرو سيستم:

نهوي-كبير: سولاريس (جهاز سن ميكرو سيستم)			
A1B2C3D4	00020004	00000000	00000000
00000044	00000001	3EBCBA2D	0004BFF0

نهوي-صغير: لينكس (جهاز إنتيل)			
00000000	00000000	D4C3B2A1	02000400
2DBABC3E	46C30500	60000000	01000000

2. ^ لضمان سلامة شفرة محمل الإقلاع في MBR، يجب ألا تتغير أبدا القيم التي في الحيود من 0DAh إلى 0DFh، إلا إذا كانت جميعها أي 6 بايت تمثل قيم 0 أو في نفس الوقت تم استبدال كامل شفرة محمل الإقلاع في MBR. (باستثناء جدول الأقسام (الممتد)). هذا يشمل تصفير [59] هذه القيم إلى 00 00 00 00 00 00 ما لم تكون الشفرة المخزنة في MBR معلومة. وأنظمة ويندوز تلتزم بهذه القاعدة.

```
0060  . . .
0070 4F 75 ED BE A3 06 EB D3 BE C2 06 BF FE 7D 81 3D [Ou..........}.=]
0080 55 AA 75 C7 8B F5 EA 00 7C 00 00 49 6E 76 61 6C [U.u.....|..Inval]
0090 69 64 20 70 61 72 74 69 74 69 6F 6E 20 74 61 62 [id partition tab]
00A0 6C 65 00 45 72 72 6F 72 20 6C 6F 61 64 69 6E 67 [le.Error loading]
00B0 20 6F 70 65 72 61 74 69 6E 67 20 73 79 73 74 65 [ operating syste]
00C0 6D 00 4D 69 73 73 69 6E 67 20 6F 70 65 72 61 74 [m.Missing operat]
00D0 69 6E 67 20 73 79 73 74 65 6D 00 00 00 00 00 00 [ing system......]
00E0  . . .
      00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F  0123456789ABCDEF
```

3. ^ القيم المختلفة عن 00h و 80h، في حقل الحالة، أصلا غير صالحة، لكن سجلات MBR الحديثة تتعامل مع بت 7 كعلم تنشيط وتستخدم هذه المدخلة لتخزين وحدة الإقلاع الفيزيائية (رقم جهاز).

4. ^ أ ب ،حقول قطاع البداية CHS ستكون مقصورة على 1+1023 أسطوانة، و 1+255 رأس [7] ، و 63 قطاع؛ ونفس هذه القيود ستكون موجودة في حقول قطاع النهاية CHS.

5. ^ أ ب ت ث ج، نطاق القطاع S سيكون من 1 إلى 63؛ ونطاق الأسطوانة C سيكون من 0 إلى 1023. ونطاق الرأس H سيكون من 0 إلى 255.[14].

6. ^ أ ب ، عدد القطاعات عبارة عن حقل مفهرس (أي عدد صحيح موجب) بالتالي، القيمة الصفر ليست صالحة، ومحجوزة ولا يجب استخدمها في مدخلات الأقسام العادية. المدخلة تستخدمها أنظمة تشغيل في ظروف معينة؛ في مثل هذه الحالات سيتم تجاهل عناوين CHS [16].

```
01B0  . . .                                           00 01  [..............]
01C0  01 00 DE FE 3F 04 3F 00 00 00 86 39 01 00 80 00  [..bb?.?...|9..|.]
01D0  01 05 07 FE FF FF C5 39 01 00 F8 AF 4E 09 00 00  [...byyA9..0 N...]
01E0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  [................]
01F0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 55 AA  [..............U.]
      00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F   0123456789ABCDEF
```

7. ^ أ، ب، ت، اقتباس: "[معظم] إصدارات نظام م.س.دوس (التي تشمل: م.س.دوس 7 [ويندوز 95]"عانت من خطأ برمجي كان يمنع إقلاع أقراص 256 رأس، لذا سنجد في تعيينات أنظمة BIOS الحديثة 255 رأس (من 0 إلى 254) RBIL [40] [39] وقيم CHS في جدول الأقسام دائما بالشكل: 63، **254**، 1023

8. ^ العنوان 0000h:7C00h هو أول بايت من 32nd كيلوبايت في RAM. علما أن، تحميل برنامج الإقلاع عند هذا العنوان يفسر لماذا 32 كيلوبايت كانت مطلوبة لخيار القرص في IBM XT. حين كان 16 كيلوبايت هو حجم RAM الأدنى في الحاسوب الأصلي IBM PC 5150.

أصل اختيار موقع الذاكرة 0x7C00 يعود إلى مواصفة BIOS وليس معالج x86. وأول ظهور لهذا الرقم كان في نظام IBM PC 5150 ROM BIOS.

الأسباب التي دفعت مطوري نظام BIOS لاختيار.هذا الموقع (1024 بايت – 32 كيلوبايت = 0x7C00) كانت كالاتي:

- من أجل حجز أكبر مساحة ممكنة لتحميل نظام التشغيل ضمن 32 كيلوبايت.
- لأن معالجات 8086/8088 كانت تستخدم العناوين 0x3FF – 0x0 من أجل المقاطعات، ومنطقة بيانات BIOS كانت تأتي بعدها.
- لأن حجم قطاع الإقلاع كان 512 بايت، ومنطقة بيانات/رصة برنامج الإقلاع احتاجت أكثر من 512 بايت.

النظام والتطبيق يستطيع استخدام 1024 بايت الأخيرة من 32 كيلوبايت. لأن بعد تحميل وتشغيل النظام، قطاع الإقلاع لا يستخدم حتى إعادة التشغيل وتخطيط الذاكرة بعد تحميل النظام سيكون كالاتي:

9. ^ في حالة وجود منطقة بيانات البيوس الموسعة EBDA، الذاكرة المتوفرة ستنتهي أسفلها.

10. ^ الأجهزة القديمة جدا قد تملك ذاكرة أقل من 640 كيلوبايت (655,360 بايت). ونظريا، 32 كيلوبايت فقط (أي حتى 0000h:7FFFh) أو 64 كيلوبايت (أي حتى 0000h:FFFFh) ستكون مضمونة؛ كما في أجهزة IBM XT المجهزة بالقدر الأدنى للذاكرة المطلوب لنظام القرص.

11. ^ روتين، مثل روتين تحريك الكتل الابتدائي primitive block move، أو مدخلات/مخرجات المستخدم، user I/O، أو تحليل دليل نظام الملفات file system directory.

12. ^ يتم تطبيق هذا عندما يقوم BIOS بمعالجة VBR، الكائن في القطاع الفيزيائي الأول من الوسيط الغير مقسم. ما عدا هذا، BIOS لا يفعل أي شيء آخر مع VBR. سجلات VBRs مصممة بهذا الشكل لأنها نشأت فقط على وسيط القرص المرن الغير مقسم. – حاسوب IBM PC 5150 لم يكن أصلا يملك أي خيار للقرص الثابت – . ونظام التقسيم الذي يستخدم MBR تم تطويره فيما بعد لإضافة أكثر من وحدة تخزين، كل وحدة تبدأ بسجل VBR خاص، ومحدد على قرص ثابت واحد، بهذا التصميم MBR يحاكي روتين إقلاع BIOS، بالقيام بنفس الأشياء التي كان سيقوم بها BIOS عند معالجة VBR وتنصيب بيئة التشغيل الابتدائية له إذا اكتشف VBR على وسيط غير مقسم.

13. ^ تعيين مؤشر التعليمة IP يكون نتيجة القفزة. يمكن تعيين قطعة الشفرة CS إلى 0 إما بالقيام بقفزة بعيدة أو تحميلها قبل القيام بقفزة قريبة. (من المستحيل على شفرة الهدف في نظام x86. اكتشاف ما إذا كانت القفزة قريبة أم بعيدة استعملت للوصول إليها [إلا إذا كانت الشفرة التي قامت بالقفز، مررت، على حدة، هذه المعلومات بطريقة ما]

14. ^ هذا ليس جزءا من العرض المذكور سابقا، ولكن نتيجة طبيعية للحالات موجودة مسبقا.

15. ^ مثال على ذلك، محرر جدول أقسام باور كويست PTEDIT32.EXE، الذي يشتغل في أنظمة ويندوز، والذي لا يزال متوفر على هذا الموقع: Symantec's FTP .

16. ^ أ، ب، في أجهزة تخزين الحاسوب، القطاع هو قسم على مسار على قرص مغناطيسي، أو قرص ضوئي. كل قطاع يخزن كمية ثابتة من البيانات، الحجم التقليدي للقطاع هو 512 بايت على القرص الثابت و 2048 بايت على الأقراص المدمجة CD-ROM/DVD-ROM. و 4096 بايت (4 كيلوبايت) [48] على الأقراص الثابتة الحديثة، وتعرف بالصيغة المتقدمة AF.

القطاع هو وحدة التخزين الصغرى على القرص الثابت. في معظم عمليات تقسيم القرص، الملفات تشغل عدة قطاعات، بصرف النظر عن الحجم الحقيقي للملف، الملفات التي لا تشغل قطاع كامل، يتم حشو بقية قطاعها الشاغر بقيم الصفر. عمليا أنظمة التشغيل تعمل بواسطة كتل من البيانات، [44] يمكنها أن تغطي قطاعات متعددة.

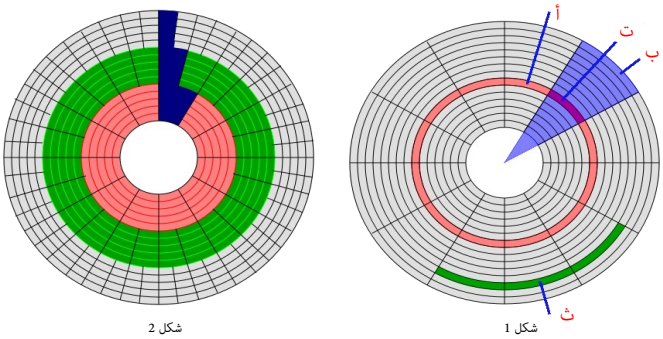
في الهندسة الرياضية، القطاع الدائري جزء من دائرة يحده نصفي قطر وقوس، (شكل 1 - ب)، أي مثل شريحة فطيرة. لذلك، قطاع القرص (شكل 1 - ت) يشير إلى تقاطع مسار مع قطاع هندسي.

في الأقراص الثابتة، كل قطاع فيزيائي مركب من ثلاث أجزاء أو قطع أساسية: ترويسة القطاع، ومنطقة البيانات، وشفرة مصححة الخطأ ECC.

- ترويسة القطاع تتضمن معلومات يستخدمها محرك الأقراص والمتحكم؛ هذه المعلومات تشمل:
  - بايتات مزامنة، وتعريف أو كشف عناوين، وعلم خطأ، و بايتات الزوجية للترويسة.

أيضا الترويسة قد تتضمن عنوان بديل يستخدم إذا كانت منطقة البيانات غير موثوقة. وكشف عناوين يستخدم لتأكد أن محرك الأقراص قد وضع رأس الكتابة/القراءة على الموقع الصحيح.

- منطقة البيانات تتضمن بيانات مسجلة للمستخدم.
- حقل شفرة مصححة الخطأ ECC يتضمن شفرات ترتكز على حقل البيانات، تستخدم في تفحص وربما تصحيح الأخطاء التي يمكن أن تقع في البيانات.



بنية القرص

أ. مسار

ب. قطاع هندسي

ت. قطاع مسار

ث. عنقود (كتلة أو كتل فيزيائية)

في الشكل 2 -تخطيط فيزيائي للقطاعات مع تقنية ZBR ولأن المسافة من المركز تزداد صعودا، عدد القطاعات في الزاوية المعطاة سيزداد من واحد (بالأحمر) إلى اثنان (بالأخضر) إلى أربعة (بالرمادي).

حجم القطاع وعدد القطاعات الإجمالي يحدد سعة القرص. ووحدة قياس سعة القرص الآن هي جيجابايت (تقريبا، مليار بايت) أو تيرابايت (تقريبا، تريليون بايت)

- القطاعات والكتل: كما ذكرنا أعلاه، القطاع يشير تحديدا إلى منطقة فيزيائية على القرص، أما الكتلة بالمطلق فهي إشارة إلى مقدار صغير من البيانات. وحسب السياق، تعبير الكتلة يملك عدة معاني. مثلا، كتلة نظام الملفات هي بنية مجردة على قطاعات القرص وربما تطوق عدة قطاعات (معظم أنظمة الملفات ترتكز على جهاز الكتل). في سياقات أخرى، قد تكون وحدة تدفق بيانات أو وحدة عملية في وسيلة مساعدة [راجع هذا المقال]. مثلا، برنامج dd في يونكس يسمح بضبط حجم الكتلة لاستخدامها أثناء التنفيذ عن طريق (خيار) معامل bs=bytes. هذا يحدد حجم مقادير البيانات الموزعة من dd، وليس له علاقة بالقطاعات أو كتل نظام الملفات.

في لينكس، يمكن معرفة حجم قطاع القرص باستخدام الأمر: `# fdisk -l | grep "Sector size"`

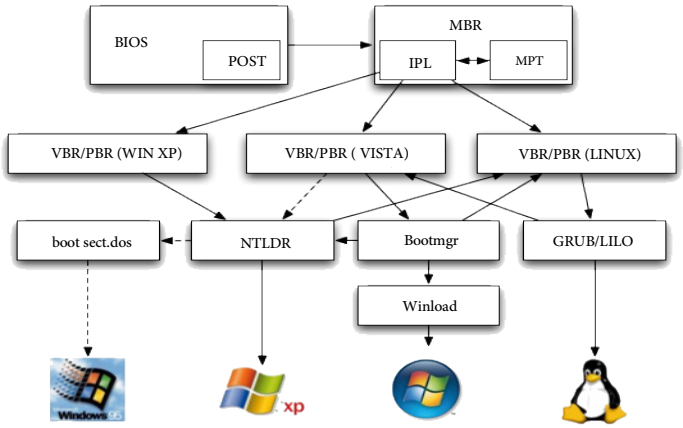
ومعرفة حجم الكتلة [راجع] باستخدام الأمر : `# blockdev –getsz /dev/sda`

- في تخطيط قرص GPT مواصفة UEFI تستخدم غالبا تعبير الكتلة Block للإشارة إلى قطاع 512 بايت أو أكثر.. بدلا من تعبير قطاع Sector. (ربما لأنها ترتكز على عنونة LBA !).

- تقنية ZBR (تسجيل البت المقسم إلى مناطق! أو محدودة المناطق!) :

كما كان الحال مع الأقراص الثابتة القديمة والأقراص المرنة، تعريف القطاع على أساس التقاطع الذي بين نصف قطر والمسار، يعني أن القطاعات الخارجية من القرص فيزيائيا ستكون أطول من تلك القريبة من المحور. ولأن كل قطاع يملك نفس عدد بايتات، القطاعات الخارجية سوف تملك كثافة بت أقل من نظيرتها الدخلية، هذا يعني استعمال غير فعال لمساحة القرص. لذلك منذ التسعينات 1990s، الأقراص الثابتة تستخدم تقنية ZBR (وتقنية شبيهة لها تدعى سرعة زاوية ثابتة – محدودة المناطق Z-CAV)، يقسم فيها القرص إلى مناطق، كل واحدة تطوق عدد صغير من المسارات المتجاورة. ومن ثم كل منطقة مقسمة إلى قطاعات وبذلك كل قطاع يملك نفس الحجم فيزيائي. ولأن المناطق الخارجية تملك محيط دائرة أكبر من نظيراتها الداخلية، فسوف تخصص قطاعات أكثر. هذا يعرف باسم نسبة البت المحدود المناطق! zoned bit rate. نتيجة استعمال تقنية ZBR عمليات القراءة والكتابة المتجاورة ستكون بشكل ملحوظ أسرع على المسارات الخارجية (توافق عناوين الكتل الأقل) مقارنة بالمسارات الداخلية، لأن بتات أكثر ستمرر تحت الرأس في كل دورة؛ نسبة هذا الاختلاف يمكن أن تكون 25% أو أكثر.

17. ^ أقسام تتضمن بيانات إعداد تصف وحدات منطقية مخزنة عند واحد 1 ميغابايت الأخير من القرص، ولا تنتمي لأي قسم، في أنظمة ويندوز أن تي.
18. ^ أ ب، رغم وجود سجل MBR في جميع الأقراص الثابتة، إلا أن شفرة الإقلاع في هذا القطاع تستخدم (في ويندوز).فقط إذا كان القرص متصل بجهاز x86 ويحتوي قسم أولي نشط



19. ^ أ ب، عمل برنامج اختبار التشغيل الذاتي أو الفحص الذاتي عند الإقلاع POST:

واختصاره بالإنجليزية POST. هو متتالية من اختبارات تشخيص في شكل برنامج آلي يفحص المكونات المادية في الحاسوب، ويبدأ عمله مباشرة بعد تشغيل الحاسوب (و BIOS هو من يشغل POST عند [59] تصفير المعالج، وأول موقع ذاكرة سيحاول المعالج تنفيذه يعرف باسم Reset vector)، بسبب عمل هذه روتينات سيلحظ المستخدم بعض التأخير بين فترتي تشغيل الجهاز وبدء تحميل النظام، هذا البرنامج لا يزال يستخدم منذ أن أنتجت شركة IBM الحاسوب الشخصي عام 1982. عادة، المهام الأساسية للبرنامج POST في BIOS ستكون كالتالي:

- التحقق من تسجيلات المعالج.
- التحقق من تكامل الشفرة نفسها في BIOS.
- التحقق من بعض المكونات الأساسية مثل الوصول المباشر للذاكرة (أي اتصال الجهاز الطرفي بالذاكرة مباشرة من دون استخدام المعالج)، والمؤقت، ومتحكم المقاطعة.
- التحقق من ذاكرة النظام الأساسية.
- تشغيل BIOS..
- تمرير التحكم إلى الامتدادات الأخرى في BIOS (إن كانت موجودة،/منصبة).
- تحديد، وتنظيم، واختيار أجهزة الإقلاع الجاهزة (محركات القرص المرن، القرص الثابت، القرص المدمج، وأية أجهزة أخرى في الحاسوب)

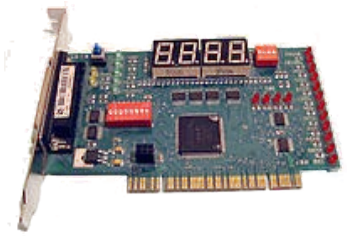
الوظائف أعلاه كانت موجودة منذ البداية في جميع إصدارات BIOS، لكن في الإصدارات اللاحقة ظهرت الوظائف التالية في برنامج POST:

- اكتشاف، وتشغيل، وتصنيف جميع نواقل النظام والأجهزة.
- توفير واجهة مستخدم من أجل تضبيط النظام.
- إنشاء بيئة النظام المطلوبة من نظام التشغيل المستهدف.

في أنظمة BIOS القديمة، POST لا ينظم أو يختار أجهزة الإقلاع، وكان فقط يحدد الأقراص المرنة أو الثابتة، التي سيحاول النظام إقلاعها بحسب ترتيبها.

أنواع الرسائل التي يصدرها برنامج اختبار التشغيل الذاتي:

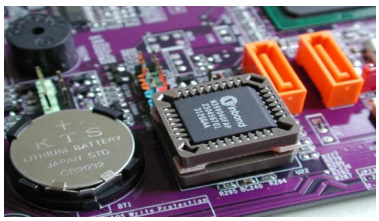
- رسائل صوتية (صفير Beeps): تعتمد رسائل الخطأ الصوتية على مكبر داخلي في الجهاز ويكون بحالتين ، إما مدمج على لوحة النظام، أو مثبت على جدار الصندوق الداخلي للحاسوب. وليس كل صوت يصدر من الجهاز يعتبر رسالة خطأ، فعندما يعمل الجهاز بشكل طبيعي ويجتاز بنجاح الاختبارات يصدر الجهاز صوت صفير قصير، بعض الأجهزة مثل كومباك يصدر الصوت مرتين.
- رسائل نصية: أحيانا تكون نصية أو رقمية يتم عرضها على الشاشة وبحسب نوع BIOS يتم تفسير هذه الأرقام لمعرفة القطعة المسببة للمشكلة. من رسائل الخطأ المشهورة:
  - 161 بطارية منتهية (تحتاج إلى بطارية جديدة في لوحة النظام)
  - 162 الإعدادات تغيرت (مثلا عند إضافة ذاكرة أو بطاقة جديدة إلى الحاسوب)
  - 301 خطأ لوحة مفاتيح.
- رسائل الرموز: يرسلها BIOS في بداية اختبارات التشخيص إلى عنوان منفذ إدخال/إخراج خاص (I/O Port) هذه الرموز لا يتم عرضها على الشاشة، ولا نستطيع استعراضها إلا بتثبيت بطاقة خاصة تحوي شاشتين رقميتين بأحد شقوق التوسعة (وأحيانا تكون هذه الشاشة مدمجة على لوحة النظام)، وهناك عدة شركات معروفة بتصنيع مثل هذه البطاقات، مثل Ultra-X و Trinitech. عند تثبيت أحد هذه البطاقات، سوف تلاحظ خلال عمل البرنامج عرض أرقام ست عشرية مكونة من رقمين، فإذا توقف الجهاز أو تجمد يمكن حينها معرفة سبب المشكلة بواسطة تلك الأرقام.



لأن نجاح البرنامج هذا يشير إلى استعادة النظام إلى حالته المعروفة، عند تعليق برنامج في النظام سيكون فصل الطاقة وإعادتها هي الطريقة المعيارية لإعادة تشغيل النظام.

20. ^ أ، ب، ت، ث، ج، ح، عمل نظام الإدخال والإخراج الأساسي BIOS:

- يستهل بتفحص العتاد بتنفيذ برنامج اختبار التشغيل الذاتي POST (أنظر أعلاه)
- يحدد جهاز الإقلاع (عادة، يكون قرص ثابت، سابقا كان يحاول قراءة قطاع إقلاع القرص المرن).
- يحمل محتويات القطاع الأول (512 بايت) على القرص في عناوين الذاكرة (7C00 - 7DFF.) عن طريق نداء المقاطعة INT 19h. مع ضبط تسجيل DL على رقم قرص تحميل MBR
- يأمر المعالج بتنفيذ شفرة MBR [21] عن طريق عمل قفزة jmp إلى الموقع 7C00. (هذا هو عنوان بداية شفرة إقلاع MBR في الذاكرة)



21. ^ أ، ب، ت، ث، ج، ح، خ، د، ذ، عمل شفرة سجل الإقلاع الرئيسي MBR:
   - تفحص جدول الأقسام باستخدام نداءات INT 13 من أجل إيجاد المدخلة الأولى التي تقبل الإقلاع.
   - تحدد الموقع الفيزيائي للقطاع إقلاع نظام الملفات عن طريق مدخلة جدول الأقسام، أو تعرض للمستخدم لائحة اختيارية بالأنظمة الموجودة (كما تفعل شفرة GRUB 2).
   - تنقل نفسها (عادة، مع قفزة بعيدة) من (7C00 - 7DFF.) إلى (0600 - 07FF.) وتستمر في عملية التنفيذ من هناك.
     - (في لينكس شفرة محمل الإقلاع LILO MBR تنتقل إلى موقع أعلى في الذاكرة عند: 0098:8A00).
   - تنقل كامل قطاع إقلاع نظام الملفات (عن طريق BIOS INT 13h) إلى عناوين الذاكرة (7C00 - 7DFF.) وتضبط DS:SI؛ (تسجيلان يشيران إلى مدخلة جدول الأقسام المحددة).
   - تأمر المعالج بتنفيذ شفرة قطاع إقلاع نظام الملفات. (مع ضبط التسجيل CS على 0 ، والتسجيل DL على رقم القرص) من هنا وصاعدا، أي إجراءات تكون تابعة لنظام الملفات.

شفرة MBR تستخدم نداء المقاطعة INT 13 لقراءة البيانات من القرص عند تشغيل الحاسوب. (حالما يتم تحميل النظام، طريقة النفاذ إلى القرص يمكن أن تتغير حسب نظام التشغيل)

شفرة MBR في أنظمة XP/2000/98 لا تنسخ أي بايت قامت بتنفيذه سابقا قبل القفز إلى موقع الذاكرة الجديد؛ الشفرة تنسخ فقط 485 بايت التي (من 7C1Bh إلى 7DFFh) إلى المواقع التي (من 061B إلى 07FF)، عوض نسخ كامل كتلة 512 بايت إلى 0600 وما يلي ذلك، كما تفعل شفرة IBM PC DOS 3.30 من أجل الخمسة والعشرون تعليمة الأولى،

شفرة سجل إقلاع XP/2000 تشبه شفرة ويندوز 98 (نظام ملفات FAT32)، لكنها بعد ذلك تتشعب إلى روتينات جديدة بالكامل. وعلى عكس، شفرة سجل إقلاع XP/98/2000، شفرة ويندوز فيستا/7 تنسخ كامل 512 بايت إلى الموقع الجديد، بدأ من 0600 .فقط الثلاثة تعليمات الأولى تشبه ما في شفرة سجل إقلاع XP/2000.

22. ^ مشكلة الالتفاف Wrap-Around:

بعض أنظمة BIOS القديمة، تفترض دائما 1,024 أو أقل عدد للأسطوانات على القرص، فتنظر فقط إلى 10 بت قاعدة لعدد الأسطوانة التي يعلن عنها القرص الثابت (2^{10} = 1,024). ونتيجة لذلك، عند استعمالها مع أعداد أكبر من 1,024، تقوم أنظمة BIOS بعملية تعداد مكافئ حتى 1.024 ثم تلتف إلى الصفر مرة أخرى وتبدأ من جديد (هذا مكافئ للقيمة النمطية "N mod 1024" حيث "N" عدد ألأسطوانات الحقيقي). مثال، إذا حاولت استعمال قرص يملك 3,500 أسطوانة، نظام BIOS سيتعرف فقط على 428 أسطوانة (3500 ـ mod 1024)، لأنه سوف يحسب حتى 1,024 ثلاثة مرات (للحصول على 3,072)، و يلتف ثلاثة مرات، ثم ينتهي مع قيمة 428 أسطوانة (3,500 ناقص 3,072). نفس الشيء يمكن أن يحدث مع نظام BIOS الذي يدعم فقط 4,096 أسطوانة. يمكنه أن ينظر فقط إلي القاعدة 12 بت. هذا يعني أن في بعض الحالات يمكنك وضع قرص ثابت بحجم 2.5 جيجابايت في نظام ثم تحصل فقط على حوالي 400 ميغابايت مساحة مستعملة. هذا النمط من الفشل كان شائع في أنظمة BIOS التي لا تدعم أكثر من 4,096 أسطوانة.

أيضا بعض أنظمة BIOS التي تدعم وظيفة الترجمة سوف تقوم بالالتفاف إذا عطلت الترجمة. وسوف يختفي المشكل إذا تم تمكين وظيفة الترجمة.

23. ^ مودولو modulo ، (دالة باقي القسمة أو تردد):
   - في الحوسبة، (اسم) عملية أو دالة تعود بباقي قسمة عدد على آخر. (راجع أكثر هذه المقالة الانجليزية في الموسوعة الحرة)
   - في الرياضيات، تعطى قيمة نمطية محددة: عملية حسابية نتيجتها بقية عملية قسمة. مثال 17 mod 3 = 2 لأن 17 مقسوم على 3 تنتج بقية 2. مودولو تستخدم في الحساب النمطي والبرمجة.

> تطبيق: ليكن لدينا عدد طبيعي m و عدد صحيح k . نعرف k (mod m)
>
> و تقرأ k تردد m . على أنه عدد طبيعي هو باقي قسمة k على m . أمثلة: باقي القسمة محصور دوما بين 0 و m - 1 :
>
> | | | | | | | |
> |---|---|---|---|---|---|---|
> | -3 (mod 8) = 5 | -39 (mod 3) = 0 | -371 (mod 8) = 5 | -26 (mod 7) = 2 | 3 (mod 8) = 3 | 35 (mod 11) = 2 | 25 (mod 5) = 0 | 25 (mod 7) = 4 |
>
> مثال أخر: الساعة عبارة عن تطبيق لدالة باقي القسمة على 12 أو 24 . 6 = (10 + 20) (mod 24)
>
> أي أنه إذا كانت الساعة الآن العاشرة صباحا فستكون السادسة صباحا بعد عشرين ساعة .(اقتباس من رياضيات الحاسوب)

24. ^ أ، ب، ت، ث، ج، ح، hard wired, Hard coded :
   - قيمة بيانات أو إجراء تم كتابته مباشرة في برنامج، غالبا في عدة أمكان، بحيث لا يمكن تعديله بسهولة.
   - (صفة) هي البيانات التي تم تضمينها مباشرة في البرنامج، حيث لا يمكن تعديلها بسهولة، خلافا للبيانات في بعض ملفات التعريف (ملفات التحكم)، أو مورد.

25. ^ كثيرا ما يوصي الناس باستخدام برنامج دوس الغير موثق FDISK /MBR في حل مشكلة MBR. في الحقيقة، هذا الأمر لا يعيد كتابة كامل MBR ولكن يعيد فقط كتابة شفرة الإقلاع (غالبا 446 بايت)، ويترك معلومات الأقسام (64 بايت) دون تغيير. لذلك البرنامج لن يساعد المستخدم إذا كانت المشكلة في جدول الأقسام. علاوة على ذلك، قد يصبح الأمر خطير إذا حاول المستخدم استعادة شفرة الإقلاع إلى حالتها الأولى، وكان سبب المشكلة فيروس قطاع الإقلاع. في هذه الحالة المعلومات الأساسية يمكن أن تكون مخزن في مكان أخر عن طريق الفيروس. والتخلص من الفيروس يعني التخلص من وسيلة الوصول إلى تلك المعلومات. (مثلا، فيروس stoned.empire.monkey يقوم بتشفير MBR الأصلي في القطاع 0/0/3. – راجع فقرة "فيروسات قطاع الإقلاع" أعلاه). على أية حال، من يريد حذف شفرة محمل إقلاع مثل LILO من قطاع MBR، ولا يعرف أن LILO يملك خيار سطر الأوامر -u ، يستطيع استخدام FDISK /MBR.

26. ^ أ، ب، ت، ث، ج (مدخلات) جدول الأقسام:
   - ترقيم مدخلات جدول الأقسام من 1 إلى 4 ليس مطلوب في MBR، لأن بعض الأنظمة والبرامج يمكن أن تخلط ترتيبها وتغير بدايتها.
   - أي قسم من الأربعة يمكن أن يحمل علم الإقلاع. ويندوز يشترط وجود قسم واحد نشيط، بينما معظم الأنظمة الأخرى لا يهمها وجود بت التنشيط في مدخلة جدول الأقسام.
   - م.س.دوس كان يعبئ جدول الأقسام بداية من النهاية. بالأخص، إذا كان هناك قسم واحد فقط، وكانت المعلومات تخزن في المدخلة الأولية الرابعة. فيما بعد أصبح FDISK DOS يبدأ من البداية. لكن أنظمة مثل UnixWare لا تزال تبدأ من النهاية. كذلك أقراص أيوميغا تكتب القسم الوحيد في قرص ZIP في المدخلة الأخيرة (وستكون موصولة بالشكل: dev/sda4 أو /dev/hdc4).

27. ^ أ، ب، ت، ث، في أجهزة الحاسوب المتوافقة مع أنظمة IBM، في حالة عدم العثور على قطاع إقلاع صالح على القرص المرن أو القرص الثابت يتم تنفيذ نداء المقاطعة INT 18. وفي أنظمة ويندوز، إذا فشلت شفرة إقلاع MBR، ستعرض إحدى الرسائل التالية:

نظام التشغيل مفقود	Missing operating system	خطأ في تحميل نظام التشغيل	Error loading operating system	جدول أقسام غير صالح	Invalid partition table

28. ^ القرص الديناميكي: تم دعمه في نظام ويندوز 2000 والأنظمة اللاحقة. القرص الديناميكي لا يستخدم جدول أقسام. ولكن يستخدم نظام قاعدة بيانات مخفية LDM للتتبع معلومات الوحدات والأقسام الديناميكية على القرص. مع القرص الديناميكي يمكن إنشاء وحدات تخزين (أقسام) تمتد عبر عدة أقراص، مثال على ذلك، الوحدات الشريطية والوحدات الممتدة، ويمكن أيضا إنشاء وحدات مع خاصية الاستجابة للخطأ [35] FT ، مثال على ذلك، الوحدات الشريطية مع تقنية بت الزوجية parity هذه الوحدات تعرف أيضا باسم RAID 5 أو الوحدات المرآوية mirrored (التي بياناتها منسوخة على قرصين أو أكثر) وتعرف أيضا باسم RAID-1 . مقارنة بالقرص الأساسي، القرص الديناميكي يملك مرونة أكبر. وهناك عدة أدوات لإدارة الأقراص الديناميكية. أنظمة مايكروسوفت التي لا تدعم القرص الديناميكي هي MS-DOS, 95/98/Me/NT/ وويندوز XP النسخة المنزلية.

29. ^ أ ب ت، الختم الزمني للقرص في أنظمة ويندوز 95B/98/98SE/Me. (6 بايت الغامضة! من الحيد 00DAh إلى 00DFh ضمن قطاع MBR).

عكس ما قد يظن البعض، هذه 6 بايت ليست هي نفسها في جميع الأقراص الثابتة، رغم أنها تظهر دائما كذلك في شكل أصفار فقط مضمنة [20] في جميع أدوات FDISK.

عند استخدام FDISK مع الخيار MBR/ على أي قرص في أنظمة 95B/98/98SE/Me، سيتم إعادة كتابة 6 بايت إلى أصفار مرة أخرى! إذن ما الذي يجعل تلك بايتات مختلفة على كل قرص!؟، نظام ويندوز نفسه يغير 4 بايت الأخيرة من 6 بايت كلما كانت أصفار. في مرحلة ما عند الإقلاع، النظام يبحث عن تلك 6 بايت في كل قرص، إذا وجدها جميعها أصفار، يغير 4 بايت الأخيرة لتعكس رقم القرص في MBR والزمن الذي تم فيه كتابة تلك بايتات، كما يوضح المثال في الجدول التالي:

بايت	مثال	محتوى	
DAh	00	2 بايت، دائما صفر	
DBh	00		
DCh	81	رقم القرص الفيزيائي (قرص أول = 80h ، قرص ثاني = 81h ...الخ)	
DDh	07	ساعات	زمن كتابة هذه 6 بايت إلى MBR
DEh	05	دقائق	الساعات، والدقائق، والثواني (بترتيب معكوس)
DFh	08	ثواني	08 05 07 => مساء 08: 05: 07

رقم القرص يعكس فقط الموقع الذي وجد فيه عندما كتب النظام ذلك البايت إلى القرص. قد تكون عدة أقراص تملك القيمة 80h في البايت DCh في MBR وهذا بذاته لن يسبب أية مشكلة. نعلم أن البايت DCh هو رقم القرص الفيزيائي، بسبب تصفير بايتات من DAh إلى DFh في MBR في القرص الثاني المتصل في نفس الوقت (الذي بالمناسبة، كان يملك القيمة 80h في بايت DCh) بعد إقلاع ويندوز ! في الساعة 7:05 صباحا، تغيرت تلك بايتات إلى: 00 00 **81** 08 05 07

- تنبيه: لا ينبغي لتلك الأنظمة أن تغير البايت الموجود عند الحيد DAh في أي MBR معياري على قرص متصل، لأن تغيير هذا البايت سوف يجعل شفرة MBR تستمر في عرض بايتات إلى أن تجد بايت الصفر! 00h الذي يدل على "نهاية" رسالة الخطأ الأخيرة [60] في شفرة MBR، (راجع الشفرة أعلاه).

بعد هذا الكشف الصغير، يبقى السؤال مطروح، لماذا وكيف استخدمت مايكروسوفت هذه بايتات في تلك الأنظمة. علما أن هذه 6 بايت ليست لها علاقة البتة بطريقة إغلاق نظام ويندوز وليست لها علاقة كذلك بتوقيع القرص [36] في ويندوز أن تي الواقع عند الحيود 1B8h – 1BBh.

في حالة إنشاء نسخة مطابقة من قرص MBR إلى قرص فيزيائي آخر، (باستخدام أحد أنظمة 95B/98/98SE/ME) ثم إعادة تشغيل النظام مع وصل القرصين بالجهاز، سوف يعلق الجهاز واحتمال ظهور الشاشة الزرقاء، بسبب وجود قطاعين للإقلاع في كل قرص متصل تحمل نفس قيم 6 بايت (الختم الزمني/رقم القرص).
الحل سيكون في تغيير الختم الزمني في أحد القرصين، بتنفيذ الأمر FDISK /MBR على القرص (الذي سيصفر تلك 6 بايت) تستطيع عمل ذلك باستخدام قرص طوارئ Win98/Me ولكن قبل ذلك يجب التأكد أن القطاع الأول للقرص لا يشغله أي مدير إقلاع أو برمجية ممتدة خاصة بنظام BIOS (أي فقط الشفرة المعتادة في Win98/Me MBR)

30. ^ أ، ب، ت، ث، ج، ح، مدخلات AAP، هذا نوع خاص من الأقسام النشيطة في MBR يستخدمها نظام DR-DOS و PTS-DOS،
اقتباس مترجم عن: ماتياس بول: "حتى الآن PTS-DOS هو نظام دوس الوحيد القادر على الاقلاع من القرص المنطقي في القسم الممتد باستخدام مدخلات تدعى بالقسم النشيط المتقدم AAP في MBR. وحتى يتوافق مع معايير دوس الحالية، النظام لن يختلف كثيرا عن عملها، لكن سوف يتطلب مدخلة للقسم خامسة خاصة قبل المدخلات الأربعة الأخرى وشفرة إقلاع موازية في MBR تفهم AAP. إذا تضمن MBR على التوقيع الخاص AAP وكانت هذه المدخلة الخاصة موجودة وتحمل علم الإقلاع، فسوف يستخدم MBR هذه المدخلة بدل إحدى مدخلات الأقسام الأربعة الأخرى. هذه المدخلة يمكن أن تشير إلى قطاع الإقلاع في أحد الأقراص المنطقية أو تشير إلى ملف بطول 512 بايت (يحمل خصائص النظام، حتى لا يتم تحريكه أثناء عملية إلغاء تجزئة القرص) في مكان ما داخل نظام الملفات، ويشكل قطاع إقلاع (مع نفس توقيع "IBM"وعنوان الحمولة وواجهة التسجيل). وعلى خلاف شفرة MBR الاعتيادية، هذه الشفرة تترجم بايت علم الإقلاع كوحدة قرص فيزيائي(80h..FEh)، بدلا من استخدامها فقط كعلم تنشيط (كما هو الحال مع 80h أو 00h في نسخ دوس الأقدم أو تعيين بت 7 أو مسحه في نسخ دوس الأحدث). بهذه الطريقة، سجل الإقلاع AAP يمكنه أيضا تحميل قطاع إقلاع من قرص آخر غير القرص الثابت الأول".

31. ^ أ، ب، ت، ث، ج، ح، خ، د، ذ، ر، ز، س، ش، طرق عنونة الكتل : المنطقية والفيزيائية CHS / LBA

بعض أنوع الأقسام تقتضي وجود أسلوب معين للنفاذ إلى القرص. خصوصا، أنواع 0Ch, 0Eh, 0Fh (التي هي نسخ LBA من 0Bh، 06h، 05h) والمرتبطة بمدخلات جدول الأقسام التي تملك قياسات القرص C/H/S = 1024/255/63 وتعتمد في النفاذ على استخدام وظائف نداء المقاطعة BIOS INT 13.

عنونة CHS: كانت في السابق وسيلة لعنونة كل كتلة بيانات فيزيائية على القرص الثابت. مع أن قيم CHS لم يعد لها علاقة فيزيائية مباشرة بالبيانات المخزنة على الأقراص باستثناء الأقراص المرنة، قيم CHS الافتراضية التي يمكن ترجمتها بواسطة إلكترونيات القرص أو BIOS لا تزال تستخدم من قبل العديد من البرامج الخدمية (خصوصا برامج تقسيم القرص، والإقلاع وفي محاذاة الأقسام) قبل ظهور أقراص IDE. كان للأقراص قياسات (geometry) توصف بثلاث ثوابت أو إحداثيات في متتابعة C, H, S وهي: عدد الأسطوانات. عدد الرؤوس. عدد القطاعات لكل مسار دائري. لكن مع ظهور عنونة الكتلة المنطقية LBA أصبح القرص يملك قطاعات يتم ترقيمها بشكل 0 ، 1 ، 2 ...إلخ

عنونة LBA: مخطط عنونة خطية بسيط يستخدم على نطاق واسع في تحديد مواقع كتل البيانات (القطاعات) المخزنة على أجهزة التخزين في الحاسوب، بالتحديد، في أنظمة التخزين الثانوي مثل أقراص SCSI، وأقراص ATA-2 المتوافقة مع معيار القرص الثابت IDE وأقراص التخزين الاحتياطية مثل Tape Drive.

في عنونة الكتل المنطقية، الكتل تقع وفقا لمؤشر عدد صحيح، مع اعتبار الكتلة الأولى LBA 0، والثانية LBA 1، إلى آخره .. وهكذا يتم تبسيط عملية تركيب القرص الثابت حيث يستعاض عن إدخال قيم أو إحداثيات CHS التي تتطلب تحديد رقم كل من الأسطوانة، والرأس، والقطاع والبيانات الأخرى برقم القطاع فقط، حيث تقوم خوارزمية LBA، المخزنة في البرنامج الثابت BIOS، بترجمة رقم القطاع هذا إلى عنوان CHS الموافق. الجدول التالي يعرض صيغ الحساب و التحويل بين CHS و LBA (مع أمثلة):

تحويل من CHS إلى LBA	حساب حجم القسم	تحويل من LBA إلى CHS
$(C \times TH \times TS) + (H \times TS) + (S - 1) = LBA$ أو $(C \times HPC + H) \times SPT + (S - 1) = LBA$	$S = (EnsSec_{LBA} - StartSec_{LBA}) + 1$	$C = LBA \div (HPC \times SPT)$ $H = (LBA \div SPT) \bmod HPC$ $S = (LBA \bmod SPT) + 1$

- C = رقم الأسطوانة، H = رقم الرأس، S = رقم القطاع. TH = إجمالي عدد الرؤوس لكل أسطوانة على القرص وفقا ترجمة BIOS. و TS = إجمالي عدد القطاعات لكل مسار على القرص.
  و LBA = عنوان الكتل المنطقية. HPC = العدد الأقصى للرؤوس لكل أسطوانة، و SPT = العدد الأقصى للقطاعات لكل مسار (التي يعلن عنها القرص الثابت)
- (مودولو "mod" تشير إلى عملية دالة باقي القسمة أو تردد، و "÷" قسمة عدد صحيح، أي خارج القسمة حيث يتم طرح أية كسور)

مثال حساب قطاع بداية القسم	مثال حساب قطاع نهاية القسم	مثال	مثال
$(0 \times 255 \times 63) + (1 \times 63) + (1 - 1) = 63$ $(0 \times 255 + 1) \times 63 + (1 - 1) = 63$	$(4 \times 255 \times 63) + (254 \times 63) + (63 - 1) = 80324$ $(4 \times 255 + 254) \times 63 + (63 - 1) = 80324$	$80262 = (80324 - 63) + 1$	$4 = 80262 \div (255 \times 63)$ $254 = (80262 \div 63) \bmod 255$ $63 = (80324 \bmod 63) + 1$

- بعد معرفة قياسات القرص الثابت، C=1024، H=255، S=63 نستخدمها مع متغيرات TH و ST .(أو HPC، و SPT) وإحداثية نهاية القطاع S=63، H=255، C=4، في إحدى الصيغ أعلاه.
- نبدأ، أولا، بحساب LBA لقطاع النهاية باستخدام صيغة التحويل (CHS-إلى-LBA) ثم نستخدم النتيجة في حساب حجم القسم.
- بما أن LBA 0 هو أول قطاع؛ حاصل ($C \times H \times S$) = مجموع LBAs) سيكون 1- مثال على ذلك: ($3348 \times 16 \times 63 = 3374784$ أي عداد LBA الأقصى هو 3374783.)

32. ^ أ، ب، ت، ث، ج، ح، القسم الممتد:

يملك مدخلة واحدة فقط من أصل 4 كحدى أقصى في جدول أقسام MBR المعياري، القسم الممتد في الحقيقية، ليس قسم ولكن حاوية تحتوي على لائحة موصولة من الأقسام المنطقية (أوالأقراص المنطقية). هذه السلسة يمكن أن تكون بطول كيفي، لكن بعض نسخ FDISK ترفض إنشاء أقسام منطقية أكبر من عدد المحارف المتوفرة للأقراص في النظام (مثلا، القرص الأخير في مايكروسوفت دوس هو 26، بينما في نوفيل دوس 7+ القرص الأخير هو 32). نوع القسم الممتد عادة يكون 05h (عنونة CHS) أو 0Fh (عنونة LBA)، (لمعلومات أكثر راجع: النسخة المراجعة من كتيب نوع القسم).

33. ^ أ، ب، ت، ث، ج، ح، القرص المنطقي:

هو وحدة تخزين منطقية (أو قسم منطقي) في القسم الممتد. وعلى عكس الأقسام الأولية في MBR التقليدي التي عددها 4 كحد أقصى ويتم تعريفها عن طريق جدول أقسام واحد فقط؛ في القسم الممتد، نجد كل EBR يسبق القسم المنطقي الذي يصفه. وفي حالة وجود قسم منطقي ثاني، يصبح EBR الأول يتضمن مدخلة تشير إلى EBR الثاني؛ وبهذه الطريقة يصبح عندنا عدة سجلات إقلاع ممتدة تشكل قائمة موصولة. هذا يعني أن المساحة المخصصة للقسم الممتد وعدد محارف القرص (في ويندوز) هي فقط التي تحدد العدد الممكن للأقسام المنطقية والتي لا تظهر في MBR. (لكن تذكر أن 2 تيرابايت هو الحجم الأقصى للقرص MBR).

34. ^ أ، ب، ت، سجل الإقلاع الممتد:

في القسم الممتد. سجل الإقلاع الممتد (EBR أو extended MBR) يشبه MBR ويستخدم جدول أقسام ممتد. (أنظر للخطاطة والطرح).(تسمية: سجل إقلاع القسم الممتد EPBR من باور كويست.)

تنبيه: (في FAT32 سجل الاقلاع (أي قطاع الاقلاع) أحيانا يدعى "سجل إقلاع ممتد" (موسع) Extended Boot Record، هذا لأن سجل الاقلاع القديم في FAT16 كان يملك قطاع واحد فقط).

جدول الأقسام الممتد:

EPT في EBR يأخذ نفس بنية جدول الأقسام العادي في MBR، لكن يستخدم فقط مدخلتان من أصل 4 مدخلات. هذه الصيغة تتكرر مع كل قرص منطقي. باستثناء القرص المنطقي الأخير في القسم الممتد. الذي يملك فقط مدخلة واحدة خاصة به، (أي 3 مدخلات لا تستخدم).

- المدخلة الأولى في جدول الأقسام الممتد للقرص المنطقي الأول تشير إلى قطاع الإقلاع الخاص بها.
- المدخلة الثانية تشير إلى EBR في القرص المنطقي التالي. (إذا كان لا يوجد أقراص منطقية أخرى، لا تستخدم المدخلة الثانية).

بنية إحدى سجلات القسم الممتدة EBR
حجم 512 بايت ضمن القسم الممتد

القطاع المنطقي 0 عند العنوان الفيزيائي C0/H1/S1

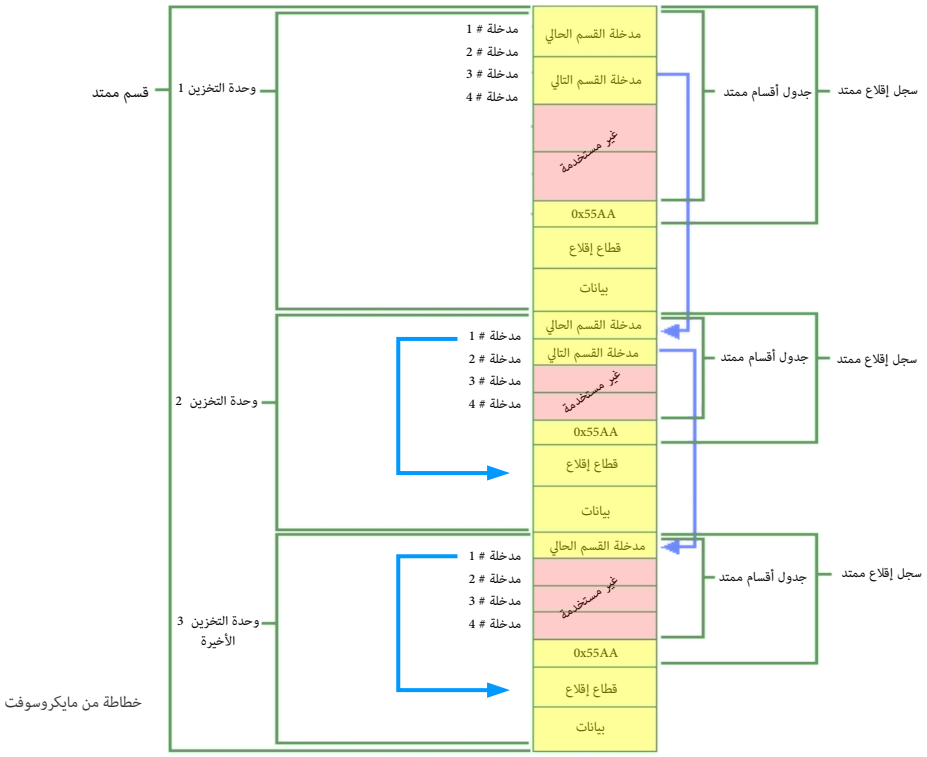
```
     00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F  0123456789ABCDEF
0000 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
*
01b0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 fe |................|
01c0 ff ff 83 fe ff ff 3f 00 00 00 b4 4b b6 0a 00 fe |................|
01d0 ff ff 05 fe ff ff f3 4b b6 0a 67 98 ee 11 00 00 |................|
01e0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
01f0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 55 aa |..............U.|
     00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F  0123456789ABCDEF
```

حشو بايت صفر [43] (هذه المنطقة قد تستغلها بعض التطبيقات المجهولة!) جدول أقسام ممتد (مدخلتان فقط) توقيع إقلاع

في نظام OS/2، برنامج fdisk يكتب بعض قيم الطول الغريبة في معلومات القسم الممتد الأخير* (ضمن القسم الممتد). على الأرجح بسبب علة في البرنامج. الذي يفشل في تحديث قيم طول القسم الممتد (الخارجي) * إذا تم إنشاء قسم أولي في المساحة الفارغة (أي المساحة الغير مشغولة من قبل القسم المنطقي) عند نهاية هذا القسم الممتد. هذا يمكن أن يترتب عنه تداخل في الأقسام.

* (تنبيه: قسم ممتد خارجي (الذي نعرفه جميعا) وآخر داخلي (قرص منطقي!) في القسم الممتد، هي من مصطلحات نظام OS/2.) (هذه المعلومات من موقع tavi و hdat2)

برنامج fdisk OS/2 لا يمكنه التعرف على نوع القسم 0Fh، لكنه يقبل أقسام دوس الممتدة التي تمتد خلف الأسطوانة 1023. لكن عندما يقوم برنامج آخر مثل Partition Magic، بتغيير نوع قسم ممتد كبير من 05h إلى 0Fh، يصبح OS/2 غير قادر على النفاذ للقسم.



باستثناء القرص المنطقي الأخير في القسم الممتد، شكل جدول الأقسام الممتد مكرر في كل قرص منطقي

35. ^ أ، ب، الاستجابة للخطأ (تحمل الأخطاء) FT. مصطلح يستخدم في علم الحاسوب للتعبير عن الخاصية التي تمكن نظاما ما من الاستمرار في العمل بشكل جيد في حال حدوث خطأ أو أكثر في أحد مكوناته (في العتاد أو البرمجيات). إذا تراجعت جودة عمل النظام، فإن هذا التراجع يكون نسبيا إلى خطورة الخطأ، مقارنة بالأنظمة التي تتوقف عن العمل تماما عند حدوث أول خطأ حتى لو كان صغيرا. بشكل رئيسي، يتم السعي وراء "تحمل الأخطاء" في حالة الأنظمة التي تتطلب تواجدية عالية أو الأنظمة الحساسة للحياة. هذا غالبا يتضمن درجة من الإضافية تشير أيضا إلى عدد الأخطاء التي يمكن أن يتحملها النظام أو المكون قبل أن تضعف العملية العادية.

36. ^ أ، ب، ت، ث، ج، ح، خ، د، ذ، بعد تنصيب وتشغيل أحد أنظمة ويندوز أن تي XP/2000/8/7، يقوم النظام بكتابة توقيع القرص إلى MBR هذه 4 بايت التي ستكون من الحيد 1B8h إلى 1BBh معروفة أيضا بالرقم التسلسلي للقرص أن تي (في الطرح أعلاه مثال على ذلك ويمكن أن تكون أية أرقام؛ لكن في أنظمة مثل ويندوز XP/2000 ستلحظ أن البايت الأول مثل الثالث والثاني مثل الرابع؛ أي تحمل نفس القيم، كما تظهر في المثال: "A8 E1 A8 E1 "، في سجلات إقلاع ويندوز أن تي أخرى، تظهر بأشكال مختلف مثلا: "87 04 88 04" و"6B 40 6C 40" و"1A 85 1A 84". والراجح أن البايت الثاني والرابع على الأقل تقريبا دائما تكون متماثلة مع احتمال وجود خوارزمية يطبقها النظام في توليد هذه الأنماط من الأرقام. رغم ذلك، هناك سجلات إقلاع في ويندوز أن تي تملك أرقام لا يمكن تميزها إطلاقا مثل"ED 19 EB BF" و"80 EF A0 FB "، ولا نعرف بالضبط الآلية التي استخدمتها هذه الأنظمة في كتابة هذه الأرقام، بخلاف التي ذكرنها سابقا.

```
01A0 . . .
01B0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 80 01 [................]
01C0 01 00 0B 7F BF FD 3F 00 00 00 C1 40 5E 00 00 00 [......?....@^...]
01D0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [................]
01E0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [................]
01F0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 55 AA [..............U.]
     00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F  0123456789ABCDEF
```

وفقا لمعلومات موقع win.tue.nl : في هذه المنطقة، برنامج GRUB كان يستخدم 4 بايت عند 0x1B8، للعنوان بداية شفرة الإقلاع المرحلة الثانية stage2، و 2 بايت عند 0x1BC لرقم الإصدارة، لكن نسخ البرنامج الأخيرة تحافظ على المنطقة 0x1B8-0x1BD. بعض الأنظمة تخزن معلومات خاصة قبل بداية جدول الأقسام. مثل نظام دي آر دوس الذي يخزن كلمة السر عند بداية الحيد 1B6h.

37. ^ في أنظمة ويندوز اللاحقة 3.XP/VISTA/2000/7/8 بايت (من 1B5h إلى 1B7h) مرتبطة باللغة الإصدارة ورسائل الأخطاء الثلاثة وعدد محارفها. وسوف تلحظ دائما هذه القيم الست عشرية في الإصدارات الانجليزية من ويندوز، مثلا في ويندوز إكس بي ستكون "2C 44 63" . هذه القيم تستخدمها شفرة MBR في عرض رسائل الأخطاء على الشاشة. لكن في إصدارات ويندوز بلغات أخرى، ستكون قيم البايت الثاني والثالث مختلفة وفقا لعدد المحارف المستخدمة في رسائل الأخطاء (ستلحظ ذلك عند تحليل شفرة رسائل الأخطاء). وسترى أن هذه 3 بايت تستخدم للإشارة إلى الإزاحة في الذكرة لأول بايت من كل رسالة خطأ يمكن عرضها على الشاشة عند الإقلاع : 072Ch، 0744h، 0763h. ولأن الشفرة التي فوق الرسائل ستكون دائما هي نفسها، الحيد الأول 072Ch لن يتغير أبدا مهما كانت اللغة المستخدمة (ومهما كان طول المحارف)، مثلا، في نسخة ويندوز XP/2000 الألمانية ستكون الرسائل في شفرة أسكي بالألمانية وستكون القيم هي: 2C 48 6E.

```
01A0 . . .
01B0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 80 01 [................]
01C0 01 00 0B 7F BF FD 3F 00 00 00 C1 40 5E 00 00 00 [......?....@^...]
01D0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [................]
01E0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [................]
01F0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 55 AA [..............U.]
     00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F  0123456789ABCDEF
```

أمثلة من ويندوز الانجليزية

ويندوز 7 / 8 / 8.1	63 7B 9A
ويندوز فيستا	62 7A 99
ويندوز XP/2000	2C 44 63

38. ^ أ، ب، ت، ث، ج، ح، خ، النمط الحقيقي، أو نمط العنونة الحقيقي: هو أحد أنماط العنونة في عائلة x86.، الأنظمة القديمة مثل دوس، كانت تستخدم هذا النمط فقط قبل ظهور النمط المحمي مع معالج 80286 . المعروف أن جميع المعالجات المتوافق مع x86 تبدأ في هذا النمط (عند الإقلاع) وتسلك سلوك معالج 8086. الذي يملك 20 خط للعنونة، أي يستطيع كحدى أقصى النفاذ إلى 1 ميغابايت من ذاكرة RAM (أي 2^{20} = 1048576 = 1024 كيلوبايت). معالج 8086 استخدم نوعان أو تسجيلان في العنونة (segment : offset)، كلاهما بعرض 16 بت وهما:

- تسجيلات القطعة. وتشير إلى عنوان في مكان ما في الذاكرة وهو من مضاعفات العدد 16.
- تسجيلات الفهرسة أو الازاحة. وتشير إلى عنوان ما بين 0 و 2^{16} = 65536 نسبة إلى عنوان يقع داخل تسجيل القطعة.

وصف		حساب العنوان الفيزيائي [39]	موقع
تشير إلى قطعة 0، إزاحة 0 = عنوان الذاكرة الحقيقي 0		0 × 16 + 0 = 0	0000h : 0000h
تشير إلى قطعة 100h، إزاحة 0Fh =. عنوان الذاكرة الحقيقي 100Fh		100h × 16 + 0Fh = 100Fh	0100h : 000Fh
لحظ: عناوين الذاكرة يمكنها أن تتداخل/تتشابك	عنوان الذاكرة الحقيقي = 20h	2h × 16 + 0 = 20h	0002h : 0000h
	عنوان الذاكرة الحقيقي = .20h	1h × 16 + 10h = 20h	0001h : 0010h

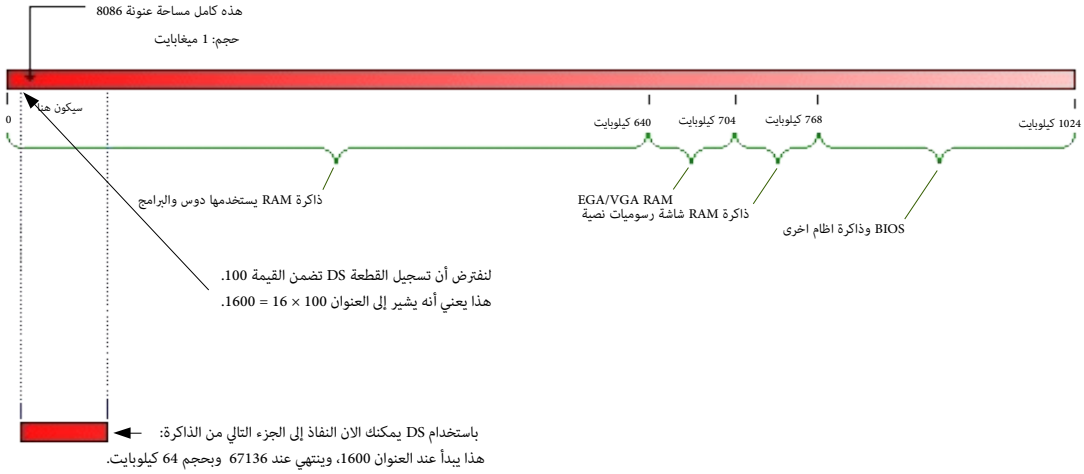
أمثلة

hex = 16h

كما ذكرنا، قيم القطعة والإزاحة 16 بت فقط. ولذلك الذاكرة القصوى المتاحة باستخدام هذا الأسلوب ستكون 0010h : 0FFFFh = واحد ميغابايت، لكن تخصيص قطعة 0FFFFh مع إزاحة أكبر من 0010h سينتج عنه التفاف (راجع أكثر فقرة النمط المحمي، خط A20 على هذا الموقع). ولأن المنطقة التي بعد 0000 : 0A000h محجوزة للنظام (من أجل الشاشة... الخ)، تتبقى فقط 640 كيلوبايت من أجل تطبيقات دوس. أيضا في هذا النمط أي برنامج سوف يستطيع قراءة/كتابة/تنفيذ الشفرة ضمن أية قطعة. ولأن في نظام تشغيل النمط الحقيقي 16-بت المعالج يرى الذاكرة في هذا الأسلوب المذكور أعلاه، أي تطبيق يستطيع القراءة والكتابة من و إلى أي جزء في الذاكرة (أي يستطيع النفاذ إلى كامل مساحة العنونة [40])، بما فيها تلك الخاصة بنظام التشغيل (أي في هذا النمط لا توجد حماية ضد البرامج التي تستطيع الكتابة فوق نظام التشغيل وتتسبب في انهياره). ولهذا السبب، نظام التشغيل في النمط الحقيقي يعتبر نظام أحادي المهام.

في النمط الحقيقي، التسجيلان CS:IP يحتفظان بمؤشر التنفيذ الحالي، و DS يحتفظ بقطعة البيانات الاعتيادية، و SS يحتفظ بقطعة الرصة. وأي تطبيق يملك أكثر من 64 كيلوبايت من قطعة الشفرة أو البيانات يجب أن يجزئها إلى قطع.

تنظيم مساحة العنونة في معالج 8086 - الحاسوب الشخصي



مثال أخر: إذا قمت بتحميل القيمة 40960 في تسجيل القطعة، سوف يشير إلى 40960 × 16 = 640 كيلوبايت باستخدام هذه القيمة يمكنك النفاذ إلى ذاكرة RAM المعيارية للرسوميات EGA/VGA التي تتضمن رسوميات الشاشة.

- VGA/EGA = محول الرسومات المحسن/منظومة العرض المرئي (معيار عرض الرسومات). Text Graphic RAM = ذاكرة شاشة الرسومات النصية
- تنبيه: تعريف 1 كيلوبايت هو 1024 بايت، و 1 ميغابايت هو 1024 كيلوبايت. [48]. الشرح الأصلي عن أنماط العنونة الأخرى موجود باللغة الانجليزية في موقع deinmeister وموقع codeproject .

39. ^ أ، ب، ت، العنوان الفيزيائي (الذي يصدر على خطوط الإشارة في اللوحة الأم) وهو نفسه العنوان المنطقي (الذي يمكنك رؤيته في برنامجك). حساب العنوان الفيزيائي سيكون بهذا الشكل:

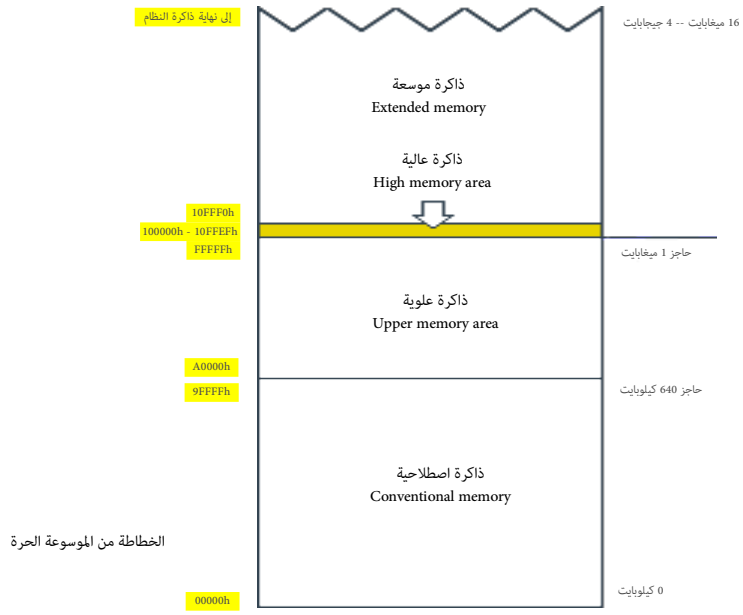
القيمة في تسجيل القطعة (عشري) × 16 + القيمة في تسجيل الإزاحة (عشري). مثال على ذلك، تحويل عنوان القطعة 0xF000:FFF0 إلى عنوان فيزيائي:

العنوان الفيزيائي = 16 × قطعة + إزاحة	16 × 61440 + 65520 # (10 × F000) + FFF0
في نظام ست عشري الضرب بـ 16 يمكن أن يتم بسهولة بإلحاق الصفر 0 إلى نهاية القطعة	= 0xF0000 + 0xFFF0 = 0xFFFF0 (1048560))

عنوان 0xFFFF0 هي 16 بايت قبل نهاية BIOS (أي 0x100000) لهذا ليس مفاجئ أن أول شيء يقوم به نظام BIOS هو القفز للخلف إلى موقع سابق في BIOS.

40. ^ فضاء/ مساحة عنونة : هي المساحة المحجوزة بالذاكرة الرئيسية للعملية الحالية. بمعنى آخر: هي ناطق من العناوين التي يمكن للمعالج أو العملية النفاذ إليه، أو عندها يمكن النفاذ إلى جهاز. المصطلح يشير إلى العنوان الفيزيائي أو العنوان الظاهري !. حجم مساحة العنونة في المعالج يعتمد على عرض ناقل عناوين في المعالج وتسجلات العناوين.

41. ^ أ، ب، تخطيط الذاكرة: في أجهزة الحاسوب الشخصي، (أو في إدارة ذاكرة نظام دوس)، ستكون ذاكرة النظام مجزئة إلى مناطق عدة مختلفة. طريقة التصميم هذه كانت وليدة للقيود نشأت في الإصدارات الأولى من أجهزة IBM PC ونظام دوس. ورغم أن البعض يلوم المسؤولين الأوائل عن هذه الطريقة المشوهة في تنظيم ذاكرة الحاسوب، إلا أن الظروف في تلك الحقبة كانت مختلفة، ففي 1981، عندما صدر أول جهاز IBM PC، كان 1 ميغابايت يعتبر حجم كبير للذاكرة. ومعظم أجهزة الحواسيب المنزلية المعروفة آنذاك كانت على الأرجح من أجهزة أبل، وكومودور 64، وكلاهما بحجم ذاكرة لا يتعد 64 كيلوبايت. في تلك الفترة، كانت بعض المدارس في الغرب، تستخدم حاسوب صغير (لكنه ضخم البنية)، بذاكرة 8 كيلوبايت! فقط، وكانت الذواكر غالية جدا. باختصار، عالم الحاسوب كان مختلف تماما عن ما نعرفه الآن. بناء على ذلك التصميم (في بداية الحواسيب الشخصية)، تم تجزئة الذاكرة الفيزيائية إلى أربعة مناطق أساسية (بعضها مجزأ أيضا) كما يظهر في الخطاطة التالية:



مناطق الذاكرة الفيزيائية في الحاسوب المتوافق مع IBM PC

المنطقة التي ما بين 640 و 1024 كيلوبايت ما زالت محجوزة حتى في أجهزة الحاسوب الشخصي الحديثة (أنظمة إنتل). لكنها ليست مرئية للبرامج (أو حتى معظم نظام التشغيل) في أنظمة التشغيل الأحدث مثل، ويندوز، لينكس، أو ماك أو.إس (ماكنتوش) تستخدم الذاكرة الظاهرية (أو الافتراضية) ولا يمكنها إطلاقا إدرك وجود عناوين الذاكرة الفيزيائية. ولذلك تعمل ضمن فضاء عنونة ظاهرية، (مساحة من العناوين الافتراضية) التي تحدد باستقلال عن عناوين RAM المتاحة.

- ذاكرة اصطلاحية أو ذاكرة تقليدية: 640 كيلوبايت الأولى (460 × 1024 بايت) (هذه ذاكرة شهيرة وسيئة السمعة في نفس الوقت). هذه المنطقة تستخدمها برامج دوس المعيارية، إلى جانب العديد من المشغلات، والبرامج المقيمة في الذاكرة، وتقريبا أي شيء يعمل تحت نظام دوس المعياري. (أول قطعة منها 64 كلوبايت تدعى ذاكرة منخفضة). هذه المنطقة تمتد من 00000h إلى 9FFFFh.
- منطقة ذاكرة علوية: 384 كيلوبايت العلوية من أول ميغابايت (مباشرة فوق الذاكرة الاصطلاحية). مجوزة من أجل أجهزة النظام وللاستخدام الخاص مثل تظليل ROM والمشغلات. هذه المنطقة تملك ثلاثة أجزاء رئيسية كل جزء بحجم 128 كيلوبايت. وتمتد من A0000h إلى FFFFFh.

ذاكرة شاشة بطاقة العرض المرئي.	128 كيلوبايت	0xA0000 - 0xBFFFF
ذواكر الجهاز BIOS ROMs، وذاكرة RAM خاصة عادة بمشاركة الاجهزة الفيزيائية	128 كيلوبايت	0xC0000 - 0xDFFFF
البرنامج الثابت BIOS ROM وذاكرة القراءة فقط، نظام Cassette BASIC.	128 كيلوبايت	0xE0000 - 0xFFFFF

- منطقة ذاكرة عالية: 64 كيلوبايت الأولى (16 بايت أقل) من ميغابايت الثاني. (تقنيا، هذه أول 64 كيلوبايت من الذاكرة الموسعة)، ويمكن النفاذ إليها عندما يكون المعالج في نمط الحقيقي، مما يجعلها مختلفة عن بقية الذاكرة الموسعة. عادة يستخدمها نظام دوس، لسماح بحجز ذاكرة اصطلاحية أكثر. هذه تحتل العناوين من 100000h إلى 10FFEFh.
- ذاكرة موسعة extended memory : تمثل كامل الذاكرة فوق HMA حتى نهاية ذاكرة النظام. تستخدم من أجل البرامج والبيانات عندما يشتغل النظام في النمط المحمي أو النمط الافتراضي في معالجات إنتل 80386 أو الاحدث، مثل ويندوز. هذه الذاكرة تبدأ من العنوان 10FFF0h إلى آخر عنوان في ذاكرة النظام. (حسب رؤيتك للأشياء،HMA تقنيا جزء من هذه الذاكرة الموسعة).
- ذاكرة موسعة expanded memory: هذه عبارة عن نظام لتبديل الضغط (في إدراة ذاكرة دوس) يوفر ذاكرة إضافية لبرامج دوس تتعد حدود ذاكرة الاصطلاحية 640 كلوبايت. هذه الذاكرة جزء من extended memory ويعينها المعالج إلى إطار صفحة الذاكرة الموسعة ويتحكم بالتعيين المدير EMM.

42. ^ أ، ب، ت، نظام نت بي أس دي NetBSD يملك التوقيع B5E1h عند الحيد 1BCh للإشارة إلى سلامة منطقة شفرة برنامج الإقلاع (Boot selector) في الحيود 190h - 1b7h. (ضمن شفرة NetBSD MBR) سابقا كانت منطقة برنامج إقلاع NetBSD في الحيود 194h - 1BBh وكلمة التوقيع كانت AA55h. لكن تم تحريك المنطقة لتجنب أي تعارض مع توقيع القرص في ويندوز أن تي، (المشكلة أن محمل الإقلاع GRUB مع أنه يحفظ هذا التوقيع، لكنه يعيد كتابة هذه المنطقة في شفرة برنامج Boot selector).

نظام NetBSD يدعم عدة شفرات للإقلاع (الطرح التالي يعرض إحداها (من ملف /usr/mdec/mbr المتوفر على الموقع الأرشيفي، لمعلومات أكثر راجع هذا الملف).

هوية الأقسام الأولية المقسمة باستخدام سجلات BSD disklabels في أنظمة بيركلي، ستكون في MBR بإحدى هذه القيم: A9h = NetBSD/. A6h = OpenBSD/. A5h = 386BSD, FreeBSD. هذه البنية تشبه تخطيط القسم الممتدة والأقسام المنطقية في مايكروسوفت دوس، وويندوز، ولينكس. أقسام BSD disklabels وأقسام مايكروسوفت دوس المنطقية في نفس القرص الثابت على الحاسوب الشخصي ستكون في أقسام أولية منفصلة.بالإضافة إلى BSD disklabels أنظمة بيركلي يمكنها أيضا الوصول إلى الأقسام الممتدة/المنطقية في نظام مايكروسوفت دوس.

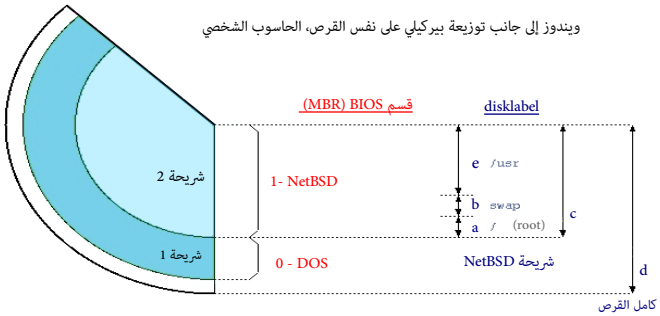
القطاع المطلق 0 (الفيزيائي) (CHS 0-0-1)

	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	0123456789ABCDEF
0000	31	c0	8e	d0	bc	00	7c	8e	c0	8e	d8	bf	1d	88	be	1d	1.....|.....
0010	7c	50	57	b9	e3	01	f3	a4	b5	03	f3	ab	cb	80	fa	8f	|PW.....
0020	7e	02	b2	80	52	52	be	1d	89	e8	4a	01	bb	94	89	8d	~...RR....J.....
0030	af	2a	00	8a	46	04	84	c0	74	0a	80	7e	00	80	75	04	.*..F...t..~..u.
0040	89	2e	84	8d	83	c5	10	83	c3	09	80	fb	b8	75	e4	b8	.....u..
0050	e1	00	c1	e0	02	89	c6	8b	b4	00	8a	85	f6	75	19	b8	.....u..
0060	2d	89	50	be	13	89	e8	13	01	5e	e8	09	01	5a	31	c0	-.P.....^...Z1.
0070	cd	13	cd	18	fb	f4	eb	fc	66	8b	6c	08	f6	06	91	89	........f.l.....
0080	04	75	43	5a	52	b4	08	cd	13	8b	44	02	50	c0	e8	06	.uCZR.....D.P...
0090	86	c4	c1	ea	08	42	f7	e2	8a	54	01	01	d0	83	e1	3f	.....B...T.....?
00a0	f7	e1	5a	83	e2	3f	01	d0	48	39	e8	74	47	bb	aa	55	..Z..?..H9.tG..U
00b0	b4	41	5a	52	cd	13	b8	65	89	72	a7	81	fb	55	aa	75	.AZR...e.r...U.u
00c0	a1	f6	c1	01	74	9c	66	89	2e	0b	89	be	03	89	b4	42	....t.f........B
00d0	5a	52	cd	13	b8	41	89	72	89	a0	00	7c	84	c0	74	03	ZR...A.r...|..t.
00e0	a1	fe	7d	3d	55	aa	b8	51	89	0f	85	75	ff	66	89	ee	..}=U..Q...u.f..
00f0	5a	e9	0c	f3	5a	8a	74	01	8b	4c	02	bb	00	7c	b8	01	Z...Z.t..L...|..
0100	02	eb	ce	10	00	01	00	00	7c	00	00	00	00	00	00	00	........|.......
0110	00	00	00	45	72	72	6f	72	20	00	**0d**	**0a**	00	4e	65	74	...Error .....Net
0120	42	53	44	20	4d	42	52	20	62	6f	6f	74	00	4e	6f	20	BSD MBR boot.No
0130	61	63	74	69	76	65	20	70	61	72	74	69	74	69	6f	6e	active partition
0140	00	44	69	73	6b	20	72	65	61	64	20	65	72	72	6f	72	.Disk read error
0150	00	4e	6f	20	6f	70	65	72	61	74	69	6e	67	20	73	79	.No operating sy
0160	73	74	65	6d	00	49	6e	76	61	6c	69	64	20	43	48	53	stem.Invalid CHS
0170	20	72	65	61	64	00	e8	03	00	be	1a	89	60	ac	b4	0e	read.....`...
0180	bb	01	00	cd	10	ac	84	c0	75	f4	61	c3	00	00	00	00	........u.a.....
0190	1c	80	b6	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
01a0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
01b0	00	00	00	00	00	00	00	00	00	00	00	00	**e1**	**b5**	00	00	.....
01c0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
01d0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
01e0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
01f0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	**55**	**aa**	.....U.

(ملف شفرة بدون تنصيب)
(NetBSD-7.0.2)
/usr/mdec/mbr

منطقة شفرة NetBSD Boot selector

بايت	شفرة
1	شفرة تفحص ضربة المفتاح الاعتيادية [49]
1	أعلام
2	مهلة (18.2 تكة في الثانية)
36	مدخلات لائحة برنامج الإقلاع للأقسام الأولية (4 × 8 بايت+ إنهاء NUL (بايت الصفر) [60]



- كلمة دوس DOS : إشارة إلى نظام تقسيم القرص وليس نظام التشغيل (نظام التشغيل قد يكون ويندوز/دوس..الخ.).
- أيضا القسم في أنظمة BIOS يدعى: شريحة SLICE (لمعلومات أكثر راجع الدليل الرسمي للتوزيعة مثل NetBSD)
- تسمية أقسام "c" و "d" كانت نتيجة حمل (أي نقل أو تكييف النظام حتى يشتغل على بيئة مختلفة) إلى منصة i386 . في معظم الحمل الأخر تستخدم "c" في تمثيل كامل القرص.

43. ^ أ، ب، ت، ث، نظرا لاختلاف أدوات إنشاء MBR لا تتوقع أن يكون دائما بايت الحشو هو الصفر. أيضا لا تتوقع أن الأداة (أيا كانت!) سوف تعيد كتابة كامل MBR، مثال: مدير RPM عندما يستبدل MBR بالشفرة IPL المعيارية [44]، سوف يكتب فقط 256 بايت (من الحيد 00h إلى الحيد FFh) زائد 37 بايت حشو صفر. أما بقية MBR ستبقى تحمل نفس القيم السابقة، أيا كانت!.

44. ^ أ، ب، ت، ث، ج، ح، خ، د، ذ، ر، ز، س، ش، ص، ض، ط، ظ، محمل ابتدائي للشفرة ! IPL / Initial Program Loader: في MBR، المساحة المخصصة لمحمل الاقلاع 446 بايت لا يمكن أن تتضمن كامل شفرة الاقلاع، لهذا تحتل هذه المنطقة شفرة ابتدائية تدعى اختصارا IPL . في أنظمة ويندوز هذه الشفرة تدل نظام BIOS إلى قطاع إقلاع ويندوز.و في لينكس من أسماء هذه الشفرة: stage1 و boot.img.وتحمل شفرة core.img أو stage1.5 التي عادة تكون في المسار الأول من القرص مباشرة بعد MBR، أو تحمل مباشرة الشفرة الرئيسية core.img أو stage2 من نظام الملفات.

45. ^ أ، ب، قسم النظام وقسم الإقلاع، هما قسمان على القرص الثابت إعدادهما ضروري لعمل النظام. لكن هناك تعرفان مختلفان لهذان القسمان:

تعريف من مايكروسوفت (يستخدم في أنظمة ويندوز) وتعريف عام مشترك تستخدمه أنظمة التشغيل الأخرى (مثل توزيعات لينكس) وسيكون بالشكل التالي:

- قسم الإقلاع : قسم أولي يتضمن محمل إقلاع، مسؤول عن إقلاع النظام.

مثال: في تخطيط أدلة لينكس FHS، (التسلسل الهرمي القياسي لنظام الملفات)، ملفات الإقلاع، مثل ملف نواة لينكس، ومخطط نظام ملفات الجذري المؤقت initrd، ومحمل الإقلاع GRUB، تكون موصولة على الجهاز /boot/. (قسم إقلاع).

وبصرف النظر عن تعريف مايكروسوفت الراديكالي (أنظر أسفل). في ويندوز أن تي، معلومات النظام، أو التطبيق المضمن في النظام تشير إلى هذا القسم باسم "جهاز الإقلاع".

- قسم النظام : قسم على القرص يتضمن دليل نظام التشغيل، ويعرف بجذر النظام.

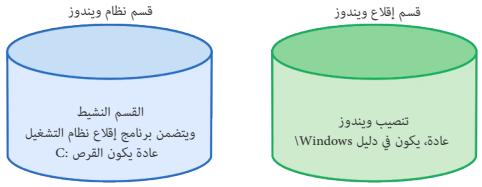
في لينكس، ملفات نظام التشغيل تكون موصولة على الجهاز " / " (الدليل الجذر). وغالبا في لينكس، قسم النظام والإقلاع يكونان قسم واحد؛ أي الدليل الجذر و /boot/ في نفس القسم.

تعريف مايكروسوفت، منذ ويندوز أن تي 3.1، (أول إصدار):

- قسم النظام : قسم أول يتضمن محمل الإقلاع، أو مدير إقلاع نظام، مسؤول عن إقلاع نظام التشغيل. هذا القسم يحتوي على قطاع الإقلاع ويحمل كذلك "علم التنشيط".
- قسم الإقلاع : قسم على القرص يتضمن مجلد نظام التشغيل، المعروف في ويندوز أن تي بجذر النظام أو %systemroot%.

في ويندوز، قسم النظام والإقلاع قد يكونان قسم واحد. أو يكونان منفصلين، رغم ذلك، قسم الإقلاع لن يتضمن برنامج الإقلاع وقسم النظام لن يتضمن جذر النظام.

قبل ويندوز 7، قسم النظام والإقلاع كنا في نفس القسم :C. بعد ويندوز 7، أصبح ينشأ قسم نظام منفصل بدون معرف ولذلك هو قسم مخفي. ويظل قسم الإقلاع يحمل المعرف/المحرف :C. هذه الإعدادات تناسب عمل برنامج BitLocker، الذي يحتاج قسم نظام منفصل بدون تشفير من أجل الإقلاع.



في حالة تنصيب ويندوز 7 بجانب فيستا. (ويندوز 7 على قرص :D و فيستا على :C). بالنسبة لويندوز 7 ، قسم النظام سيكون :C، والإقلاع :D. أما في نظام فيستا، قسم الإقلاع والنظام كلاهما سيكون :C.

46. ^ أ، ب، ت، ث، تاريخ الأقراص الثابتة معروف بمشاكل حواجز وقيود التخزين : مثل على ذلك مشكلة حاجز 137 جيجابايت التي سببها رقاقة BIOS في الأجهزة التي تستخدم قيم 28-بت فقط مع هذه القيمة (FFF FFFh أو 128 جيجابايت) بدلا من 32-بت المستخدمة في جداول الأقسام! لمعلومات أكثر عن الجدول التالي راجع هذا الموقع الانجليزي.

حواجز		حواجز	
الأسطوانة 1024	504 MiB / 528 MB	واجهة نداء مقاطعة Int 13	7.88 GiB / 8.46 GB
الأسطوانة 4096	1.97 GiB / 2.11 GB	ويندوز 95	29.8 GiB / 32.0 GB
حجم قسم FAT16	2.00 GiB / 2.15 GB	الأسطوانة 65,536	31.5 GiB / 33.8 GB
الأسطوانة 6,322	3.04 GiB / 3.26 GB	ويندوز 98/98SE	64GB
خلل في نظام فينكس Phoenix BIOS 4.03 / 4.04	3.05 GiB / 3.28 GB	قيود FAT 32	124.55 GiB / 127.53
الأسطوانة 8,192	3.94 GiB / 4.22 GB	قيود واجهة ATA	128 GiB / 137 GB
مشكلة 240 رأس وقيود واجهة Int 13	7.38 GiB / 7.93 GB		

47. ^ أ، ب، ت، ث، ج، ح، خ، د، ذ، ر، محاذاة الأقسام: عادة الأقسام تبدأ و تنتهي على حدود الأسطوانة. و VBR يقع عند (الرأس 0، القطاع 1) في جميع الأقسام الأولية، لكن بسبب احتلال MBR القطاع الأول (CHS 0/0/1) سجل VBR في القسم الأولي الأول، يبدأ على المسار التالي (أي الرأس 1 من الأسطوانة الأولي CHS 0/1/1)، في حين أن الأقسام الأولية الأخرى ستبدأ على التوالي بالشكل xxx/0/1..

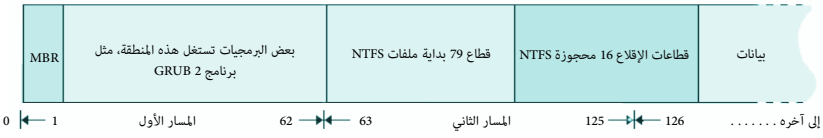
ملاحظة: سواء كان القسم الأول يتضمن لينكس أو ويندوز، المسار الأول من القرص الثابت (عادة 63 قطاع) تكون شاغرة باستثناء القطاع الأول MBR وجميعها بدون تهيئة (أي بدون نظام ملفات). في ويندوز 7 عدد القطاعات الشاغرة عند بداية القرص الثابت ارتفع إلى 2048 قطاع. علما أن MBR مع كامل المسار الأول ليس جزء من أي نظام ملفات (أو قسم نظام). ولا حاجة للقلق على ويندوز عند تنصيب محمل الاقلاع في تلك المنطقة أو في MBR. إلا إذا كانت تلك المنطقة تتضمن برامج مفيدة. لا يريد المستخدم حذفها. أيضا الأقسام المنطقية في القسم الممتد يجب أن توفر مساحة من أجل جدول الأقسام الممتد EPT. لهذا السبب، نجد أن الأقسام المنطقية تتبع نفس المخطط مثل القسم الأولي الأول. لمعلومات أكثر راجع هذه المواضيع (في الموسوعة الحرة الانجليزية).

عادة القطاع الأول في القسم الأول يبدأ عند القطاع 63 و في أنظمة التشغيل الحديثة عند القطاع 1024 أو 2048. (عند استخدام برامج التقسيم الحديثة) مثال:

```
# sudo fdisk -lu
[sudo] password for adam:
Disk /dev/sda: 160.0 GB, 160041885696 bytes
255 heads, 63 sectors/track, 19457 cylinders, total 312581808 sectors
Units = sectors of 1 * 512 = 512 bytes
Disk identifier: 0x000ba675

Device    Boot      Start         End      Blocks   Id  System
/dev/sda1            63    42331274    21165606   83  Linux
/dev/sda2      42331275   209905289    83787007+  5   Extended
/dev/sda3   *  209905290   312576704    51335707+  7   HPFS/NTFS
/dev/sda5      42331338   204266474    80967568+  83  Linux
/dev/sda6     204266538   209905289     2819376   82  Linux swap / Solaris
```

- منطقة القطاعات المخفية: كما ذكرنا سابقا، هذه القيمة من المفترض أن تكون عدد القطاعات الفيزيائية على القرص التي تسبق القطاع الأول في وحدة التخزين؛ هذا يفسر لماذا مدخلة القسم الأول في جدول أقسام القرص تملك القيمة 63 (00 00 00 3F) هذه القيمة ستكون قي حالة القسم الأولي الأول في القرص الثابت (من 0-1-0 CHS إلى 0-0-63 CHS) أو القرص المنطقي الأول في القسم الممتد (لأن كل وحدة تخزين في القسم الممتد يسبقها سجل إقلاع ممتد خاص)، لكن في حالة حساب الأقسام الأولية: الثاني أو الثالث أو الرابع أو القرص المنطقي الثاني أو أكثر في القسم الممتد، القيمة ستكون عدد قطاعات جميع الأقسام الأولية التي قبل وحدة التخزين أو عدد القطاعات من بداية القسم الممتد بالنسبة للقرص المنطقي. لذلك هذه القيمة تتفاوت فقط إذا كان القسم الأولى هو الثاني أو الثالث أو الرابع وليس الأول. علما أن في ويندوز فيستا/7 عدد القطاعات المخفية / المحجوزة لأول قسم ارتفع إلى 2048 قطاع بدل 63. وكذلك الحال في الأنظمة الأخرى.



48. ^ أ، ب، ت، ث، ج، ح، رغم أن 1024 بايت (الثنائية) =1 كيبايت، و 1000 بايت (العشرية) =1 كيلوبايت غالبا ما تستخدم تسمية كيلوبايت للإشارة إلى القيمة 1024

تحديد حجم القطاع المستخدم على القرص المنطقي والفيزيائي

```
$ cat /sys/class/block/sdX/queue/physical_block_size
$ cat /sys/class/block/sdX/queue/logical_block_size
```

مضاعفات البايت (ترجمة خاصة)

ثنائي					عشري		
معيار مجلس JEDEC (المجلس المشترك لهندسة الإلكترونيات)		معيار لجنة IEC (اللجنة الكهروتقنية الدولية)		قيمة	مترى Metric (سوابق النظام الدولي للوحدات)		قيمة
كيلوبايت	KB	كيبايت	KiB	1024	كيلوبايت	kb	1000
ميغابايت	MB	ميبايت	MiB	1024^2	ميغابايت	MB	1000^2
جيجابايت	GB	جيبايت	GiB	1024^3	جيجابايت	GB	1000^3
---		تيبايت	TiB	1024^4	تيرابايت	TB	1000^4
---		بيبايت	PiB	1024^5	بيتابايت	PB	1000^5
---		إكسبايت	EiB	1024^6	إكسابايت	EB	1000^6
---		زيبايت	ZiB	1024^7	زيتابايت	ZB	1000^7
---		يوبايت	YiB	1024^8	يوتابايت	YB	1000^8

49. ^ شفرة تفحص ضربة المفتاح أو شفرة مسح (scancode / scan code): هي بيانات ترسلها معظم لوحات المفاتيح في الحاسوب من أجل الإعلان عن هوية المفتاح المضغوط. وتتكون من رقم أو متتالية أرقام مخصصة لكل مفتاح. في الأجهزة المتوافقة مع أنظمة اي بي ام، حجم هذه الشفرة من 1 بايت إلى 3 بايت. معظم المفاتيح المحرفة تملك شفرة أحادية؛ المفاتيح التي تؤدي وظائف خاصة تملك شفرة من 2 أو 3 بايت، عادة تبدأ مع البايت E0، أو E1، أو E2. أيضا، بعض المفاتيح التي ترسل شفرة أطول، تحاكي عمليا سلاسل مفاتيح لتسهيل معالجتها ضمن أنواع مختلفة من البرمجيات.

لوحات المفاتيح في الحاسوب الشخصي تدعم ثلاث مجموعات من هذه الشفرة (منذ لوحة مفاتيح PS/2)، أكثرها انتشار هي.XT.

شفرات scancodes(IBM PS/2) المتوافقة مع أنظمة اي بي ام

مفتاح (أمثلة)	مجموعة 1 (IBM PC XT)		مجموعة 2 (IBM PC AT)		مجموعة 3 (IBM 3270 PC)	
	ضغط	إفلات	ضغط	إفلات	ضغط	إفلات
A (محرف عادي)	1E	9E	1C	F0 1C	1C	F0 1C
(محرف) الرجوع إلى السطر Return / Enter (لوحة المفاتيح الرئيسة)	1C	9C	5A	F0 5A	5A	F0 5A
مفتاح Enter (لوحة المفاتيح الرقمية)	E0 1C	E0 9C	E0 5A	E0 F0 5A	79	F0 79
مفتاح ويندوز على اليسار	E0 5B	E0 DB	E0 1F	E0 F0 1F	8B	F0 8B
مفتاح ويندوز على اليمين	E0 5C	E0 DC	E0 27	E0 F0 27	8C	F0 8C

50. ^ في عام 1984 عندما ذكرت إحدى النشرات العلمية الأمريكية إمكانية إصابة الحاسوب بالفيروس، أنذاك لم يأخذوا خبراء! الحاسوب الأمر بجدية. وبعد فترة ظهر فيروس براين عام 1986. ولم يعلم أحد أن فيروس براين الحميد ! كان فقط التمهيد لبدأ عصر الفيروسات المدمرة والمكلفة للشركات والأفراد. في ديسمبر/كانون الأول في ألمانيا، ظهر فيروس virdem (برنامج متكرر) وكان أول فيروس يصيب ملفات دوس في الأجهزة المتوافقة مع أنظمة أي بي أم. ثم بعد عام، ظهر فيروس فينا في ديسمبر/كانون الأول 1987، ثم فيروس ليهاي في نوفمبر/تشرين الثاني، وكان يصيب الشفرة التنفيذية. بالإضافة إلى فيروسات أخرى مجهولة ظهرت قبل و بعد ذلك التاريخ، وحتى وإن تشابهت هذه الفيروسات، تظل تعتمد يشكل كبير على البنية المعمارية للحاسوب (مثل التبعية للمعالج) والرتين المحمل، وتخطيط الذاكرة. هذا يفسر لماذا فيروس جهاز أبل 2 لا يصيب الحاسوب الشخصي، والعكس صحيح. رغم ذلك كتابة شفرة فيروس ثنائية من أجل بنيتان مستقلتان ممكن جدا، مثال على ذلك، فيروس بيالف متعدد المنصات، الذي ظهر في مارس/آذار 2001.

عادة، شفرة الفيروس تملك من 500 إلى 1000 تعليمة، معظمها من أجل آلية التكرار الذاتي (أو التناسخ). أما تأثير الفيروس (أو الحمولة) فتحتل فقط جزء صغير من الفيروس، ويمكن لأي مبرمج عادي تغيرها بسهولة بتعديل الفيروس الموجود. مثلا، فيروس نيوزيلندا يملك 50 طفرة (تعديل)، معظمها تغييرات بسيطة على رسالة 'Your PC is now Stoned".

جدير بالملاحظة، أن تخريب البيانات على القرص الثابت يمكن إنجازه بواسطة 5 تعليمات فقط (في لغة التجمعية). وتعديل فيروس مشهور لعمل ذلك يستغرق دقائق فقط باستخدام DEBUG.

قد يكون كاتب الفيروس مبتدئ، أو طالب، أو مجرم، أو مخرب، أو لص، أو تنظيم إرهابي مثل الكيان الصهيوني (في فلسطين المحتلة) أو دولة مارقة مثل روسيا أو أمريكا (الصهيونية المسيحية)، مثال على ذلك فيروس ستوكسنت الذي أصاب بـأجهزة البرنامج النووي الإيراني، طبعا دوافع هؤلاء تختلف. والملاحقة القضائية لهؤلاء في الغالب مستحيلة لطبيعة هذه الجهات، وكذلك لغياب القانون الخاص بذلك وأيضا لوجود اختلاف في تعريف هذا النوع من الجرائم في معظم بلاد العالم.

51. ^ أ، ب، ت، ث، ج، ح، خ، د، الفيروسات المقيمة في الذاكرة TSR:

الفيروسات المقيمة في الذاكرة، بعد أن تنصب نفسها في الذاكرة (تتحكم في الجهاز) تخطف[54] مقاطعة واحدة أو أكثر (من جدول [56] IVT، عادة تكون Int 13) وتنقل العدوى إلى الكائنات الأخرى وفق شرط معين، (كما يفعل فيروس كاسكاد عندما يحاول المستخدم تنفيذ تطبيق معين، أو فيروس براين عندما يحاول النفاذ إلى قرص مرن بمراقبة عمليات الولوج للقرص *) رغم أن قطع التيار الكهربائي عن الجهاز سوف يزيل هذه الفيروسات من الذاكرة؛ إلا أن عملية إعادة التشغيل قد لا تجدي. لأن بعض الفيروسات مثل فيروس بيل تعترض مقاطعة مفاتيح Ctrl-Alt-Del.

* ولوج للقرص تعني: ولوج الذاكرة إلى قرص الحاسوب الذي يخزن المعلومات.

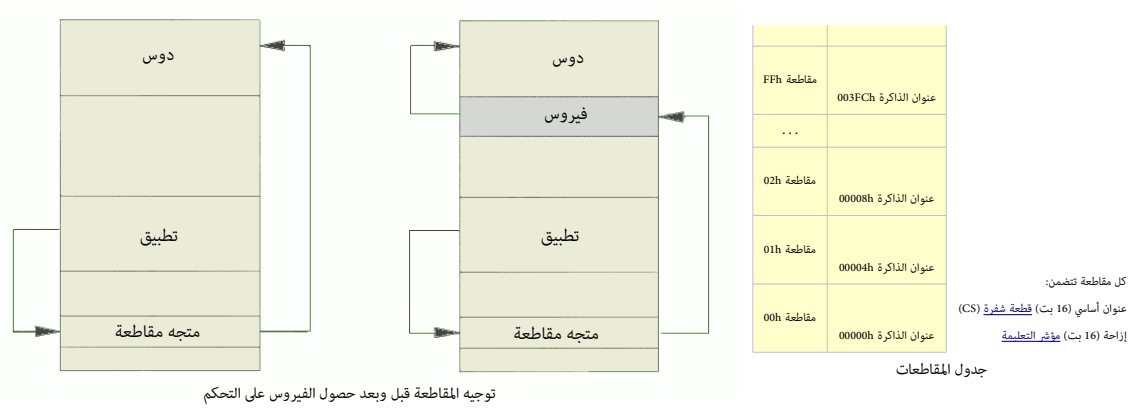
52. ^ في بعض اللوحات الأم، يمكن لنظام BIOS أن يمنع تعديل قطاعات الإقلاع. وهذا مفيد في صد فيروسات قطاع الإقلاع. لكن، عادة هذه الوظيفة في BIOS تكون معطلة، كي تستطيع أدوات تشفير وتقسيم القرص العمل، لكن إذا كان المستخدم لا يحتاج إلى تلك الأدوات مع القرص، يمكنه إعادة تمكين مرة أخرى هذه الميزة في إعادات BIOS.

53. ^ أ، ب، روتكيت: مجموعة برامج تستعمل لإخفاء عمليات نشيطة أو في طور الإنجاز على الحاسوب أو إخفاء بيانات نظام ملفات بالنسبة لنظام تشغيل. روتكيت تستخدم بشكل متزايد من قبل

البرمجيات الخبيثة لمساعدة المهاجمين للوصول إلى نظام مع تجنب الكشف من طرف المستخدم أو أداوت المستخدم . كلمة روتكيت مركبة من كلمتين: روت Root و تعني جذر و كيت kit هو ممثل من تجميع عدة أجزاء عند تجميع كلمتين نتحصل على روتكيت و يعني برامج نصية أو سكريبت التي تهاجم جذور المعالج أي النواة.

54. ^ أ، ب، ت، خطف أو الموضع الإضافة في الروتين. تكنيك في البرمجة يستعمل ما يسمى بالخطاطيف لعمل سلسلة من العمليات كمدبر للحدث. بحيث أنه بعد تحقق الحدث المدبر فإن سريان التحكم يتبع السلسلة في صورة محددة.

55. ^ أ، ب، ت، جدول المقاطعات: تطبيقات نظام دوس تستخدم المقاطعات البرمجية في تواصلها مع نظام التشغيل. عناوين القفزة ستكون مخزنة في جدول المقاطعات الواقع في بداية الذاكرة (أنظر للشكل). نظام التشغيل يقوم بإعداد هذا للإشارة إلى العناوين الحالية وفقا لإصدارة دوس. عندما تصدر مقاطعة من تطبيق، تحدث قفزة إلى العنوان المحدد مسبقا. إذا غير الفيروس عنوان أو أكثر من هذه العناوين، عند ذلك أية قفزة إلى نظام التشغيل يستطيع الفيروس إعادة توجيهها، وتوظيفها (أنظر للشكل). إمكانية تعديل جدول المقاطعات بهذه السهولة سهل ظهور وانتشار الفيروسات المتخفية.



جدول المقاطعات

توجيه المقاطعة قبل وبعد حصول الفيروس على التحكم

مثال على ذلك، فيروس براين إذا كان نشيط في الذاكرة، ثم طلب تطبيق معين من نظام التشغيل قراءة مضمون قطاع الإقلاع على القرص (حيث يكمن الفيروس)، الفيروس سيرد على الطلب بإرسال محتويات قطاع الإقلاع الأصلي، بدلا من المضمون الحقيقي (أي شفرة الفيروس). فيروس براين يستطيع فعل ذلك عن طريق تغيير جدول المقاطعات (بإقحام نفسه في الجدول).

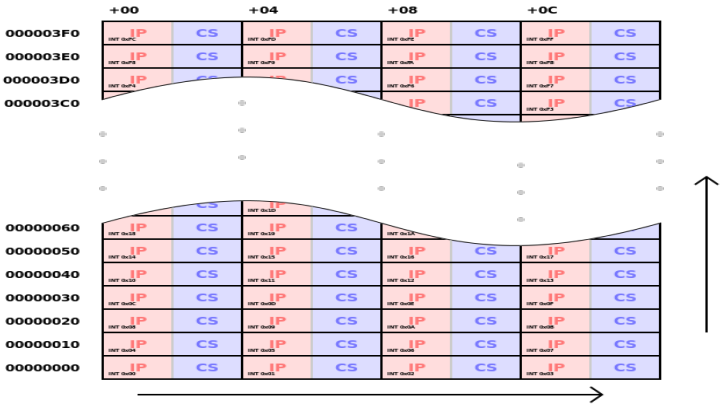
هناك عدة فيروسات تستخدم تقنية التخفي هذه. مثل فيروس 4K الذي يعترض حوالي 18 وظيفة من مقاطعة دوس 21h، تشمل إيجاد أول ملف في الشرط 11h، وإيجاد الملف الثاني في الشرط 12h، وفتح الملف 3Dh، وغلق الملف 3Eh. أيضا من بين الأشياء التي يستطيع الفيروس عملها، خصم 4096 من طول أي ملف مصاب يعرض بواسطة الأمر DIR.، بل يذهب أبعد من ذلك، عند محاولة التطبيق قراءة الملف المصاب، الفيروس سيزيل العدوى من الملف. ثم يعاود إصابة الملف عند إغلاقه. لذلك مضادات الفيروسات لن تستطيع اكتشاف 4K في الملفات المصابة إذا كان الفيروس نشيط في الذاكرة.

مثال آخر، فيروس حوشي يقوم بإخفاء مضمون قطاع الإقلاع المصاب باعتراض مقاطعة نظام البيوس Int 13 وتوجيه أي تطبيق يحاول قراءة القطاع المصاب إلى مضمون قطاع الإقلاع الأصلي.

الفيروس أيضا يعترض مقاطعة لوحة المفاتيح 9h، وينجو من علمية إعادة التشغيل باعتراض مقاطعة مفاتيح Ctrl-Alt-Del. لذلك المستخدم سيحتاج إلى قطع التيار الكهربائي عن الجهاز لإزالة هذه الفيروسات من الذاكرة. وتنصيب برنامج مضاد للفيروسات مناسب، وتفعيل حماية الأقراص المرنة من الكتابة فوقها.

أنواع المقاطعة	وصف
0h-1Fh	مقاطعات نظام البيوس BIOS
20h-3Fh	مقاطعات نظام دوس
80h-F0h	بيسيك ROM BASIC
F1h-FFh	غير مستخدمة

56. ^ أ، ب، ت، ث، متجه المقاطعة أو IVT : مكان في ذاكرة الحاسوب يخزن عناوين روتينات ثانوية. وكل مقاطعة تتضمن: عنوان أساسي (16 بت) قطعة شفرة (CS)، إزاحة (16 بت) مؤشر التعليمة (IP). جدول متجهات المقاطعة IVT (وتدعى أيضا عناوين نداءات المقاطعة) عبارة عن بنية بيانات مرتبطة بلائحة من روتينات المقاطعة ISR مع لائحة من طلبات المقاطعة، وكل مدخلة في متجه مقاطعة تشكل عنوان روتين مقاطعة ISR.



جدول متجهات المقاطعة في أنظمة x86

الخطاطة من الموسوعة الحرة

57. ^ أ، ب، ت، أغطية نظام البيوس DDO، تقنية برمجية تسمح لنظام BIOS (نظام الإدخال والإخراج الأساسي) الذي لا يدعم LBA (عنونة الكتل المنطقية،) النفاذ إلى الأقراص الأكبر من 504 ميغابايت. هذه التقنية استمر استخدمها مع هذا النوع من المشاكل حتى امتداد LBA-48. هذه التقنية تتجاوز مشغل متحكم القرص الثابت في BIOS في ذاكرة RAM.
للسماح بالوصول إلى كامل حجم القرص الثابت، البرمجية يجب أن تحمل قبل أن تحاول البرامج الأخرى الوصول إلى الأجزاء العليا من القرص. وللتأكد من تحميل هذا الامتداد في وقت مبكرا ينبغي تعديل سجل الإقلاع الرئيسي على قرص الإقلاع وتنصيب هذه البرمجية في بداية القرص. البديل لهذه البرمجية، ترقية نظام BIOS أو تنصيب بطاقة متحكم EIDE مع نظام BIOS المناسب.

58. ^ أ، ب، ت، ث، ج، ح، خ، د، ذ، ر، LSB/MSB:
- بت أقل أهمية LSB / بت أدنى Low bit
بت ذو القيمة الأدنى أو الخانة الأقل أهمية LSB أو least significant bit: هو بت على اليمين، يسمى بت منخفض low bit (بت أدنى ضمن البايت) بمعنى صفر بت (لأن ترقيم البت يبدأ من 0 ثم يزداد بواحد في كل موضع بت لاحقا)، وهو بت رقم ثنائي يعطي رقم أحاد، وهو أيضا البت الأخير أو بت أقصى اليمين في الكتابة العادية.
  - المعنى الموازي للمذكور أعلاه هو البايت الأقل أهمية Least significant byte، (نادر) وهو بايت أو ثمانية (octet) في موضع رقم متعدد البايت يملك أقل قيمة ممكنة. ويسمى بايت منخفض (بايت أدنى) low byte.لكن إذا كان المعنى في السياق غير واضح، ينبغي ذكره تجنبا للخلط مع معنى least significant bit.
- بت أكثر أهمية MSB / بت أعلى High bit
البت ذو القيمة الأعلى،أو الخانة الأكثر أهمية MSB : هو آخر بت على اليسار، يسمى high bit (بت أعلى ضمن البايت) بمعنى n-1 بت في رقم بت ثنائي، بت أكثر أهمية (2^(n-1)). بت أول أو بت أقصى اليسار في الكتابة العادية.
  - المعنى الموازي للمذكور سابقا هو بايت القيمة الأكثر أهمية Most significant byte، وهو بايت أو ثمانية (octet) في موضع رقم متعدد البايت يملك أكبر قيمة ممكنة. ويسمى بايت أعلى High byte. لكن إذا كان المعنى في السياق غير واضح، ينبغي ذكره حتى تتجنب الخلط مع معنى Most significant bit.

59. ^ أ، ب، ت، التصفير أو الاستبداء Reset في الحواسيب أو أنظمة نقل البيانات، يعني إزالة أية أخطاء أو أحداث وإعادة النظام إلى حالته الطبيعية الأولى.ويتم ذلك عادة استجابة لخطأ عندما يستحيل المضي قدما في النشاط أو لا يرغب في ذلك.

60. ^ أ، ب، ت، ث، ج، سلسلة محارف متبوعة ببايت صفر وحيد، تستخدم في عدة لغات برمجة. وتعرف بسلسلة منتهية بصفر Null-terminated string.

61. ^ أ، ب، ت، ث، ج، تنويعات قرص سوبر فلوبي superfloppy:
الوسائط المتعددة القابلة للإزالة (للفصل) التي بدون تهيئة MBR أو GPT تدعى باللغة الانجليزية "superfloppy". يتم التعامل مع كامل الوسيط باعتباره قسم واحد. وحتى وإن تضمن الوسيط على MBR، سيكون هناك دعم لقسم واحد فقط. هناك بعض الاختلاف بين الوسيط المقسم باستخدام MBR ووسائط superfloppy. وسائط سوبر فلوبي superfloppy تشمل الأقراص المرنة، خرطوشات قرص الجاز، القرص المغناطيسي البصري، دي في دي (قرص الفيديو الرقمي)، وقرص الذاكرة المدمج، الأقراص الثابتة على النواقل الخارجية مثل SCSI أو IEEE 1394 لا تعتبر وسائط قابلة للإزالة.

62. ^ Option ROM: برنامج ثابت يتم استدعاه من قبل BIOS، مثال على ذلك وحدة العرض التي تتحكم في جهاز إقلاع يمكن أن يتضمن برنامج ثابت يستخدم لربط الجهاز بالنظام حالما يتم تحميل Option ROM. أشهر مثل على Option ROM هو Video BIOS الموجود في بطاقة العرض المرئي في الحاسوب الشخصي. (أي نظام BIOS في بطاقة العرض المرئي) هذا النوع الخاص من Option ROM يتم تحميله في وقت مبكر عند الإقلاع من أجل عمل الشاشة أثناء عمليات مثل POST (الفحص الذاتي عند الإقلاع).قبل تحميل مشغل الفيديو المخصص للشاشة.

63. ^ الحواسيب التي ترتكز على أنظمة RISC لا تملك حد معين لحجم أقسام النظام أو الإقلاع.

64. ^ وحدة قرص، drive unit قد تعني hard disc، أو harddisk

65. ^ أنواع جدول الأقسام، القديمة والحديثة والخاصة

جدول أقسام آي إكس المستخدم في أنظمة آي بي إم إيه آي إكس الاحتكارية	Advanced Interactive eXecutive / aix	AIX
جدول أقسام RDB في أجهزة وأنظمة أميغا	Amiga rigid disk block / RDB	AMIGA
جدول أقسام BSD disklabel، في توزيعة برمجيات بيركلي	BSD disklabel	
ترويسة وحدة تخزين القرص في أنظمة سيليكون غرافيكس SGI	Disk volume header / dvh	sgi
جداول أقسام GPT، جزء من مواصفة UEFI (خليفة MBR و BIOS في الأجهزة الحديثة)	GUID Partition Table / GPT	
جدول أقسام أبل (قبل GPT) مخطط APM في أجهزة أبل، (PowerPC) التي يمكنها الإقلاع فقط من APM.	Apple Partition Map / Mac / APM	
جدول أقسام MBR معظم أجهزة الحاسوب الشخصي تستخدم هذا المخطط	Master Boot Record / MBR / Msdos / Intel	
جدول أقسام PC-98. في أجهزة حاسوب إن إي سي Intel x86، التي كانت شائعة في يابان	Pc98 / NEC PC-98	PC-98
جدول أقسام صن في أنظمة صن ميكروسيستمز	sun	Sun
جدول أقسام هيوماكس في أجهزة شركة هيوماكس (كوريا الجنوبية)	humax	HUMAX
جدول أقسام إكس بوكس من أجل أجهزة التخزين في إكس بوكس	Xbox	XBOX
النفاذ إلى القرص الخام (loopback-style)	raw disk access	loop

مراجع

1. ^ دنيس هو،Denis Howe (مايو/أيار 19، 2009). "master boot record " سجل الإقلاع الرئيسي". *FOLDOC*. جدد في مايو/أيار 2، 2015.
2. ^ أ ب ت ث، "نظام ويندوز يدعم أقراص أكبر من 2 تيرابايت !". مايكروسوفت. تاريخ 2013-06-26. جدد في 2013-08-28.
3. ^ أ ب. سيدوري دانيال Sedory, Daniel B (2004). "البايتات الغامضة (بايتات الختم الزمني للقرص) في سجل إقلاع أنظمة ويندوز 95B, 98, 98SE ". جدد في 2012-08-25.
4. ^ لوكاس مايكل Lucas, Michael (2003). كتاب "*Absolute OpenBSD: Unix for the practical paranoid*". صفحة 73. ISBN 9781886411999. جدد في 2011-04-09. اقتباس: "جميع أنظمة التشغيل تملك أداوت لإدارة أقسام سجل الإقلاع. ولكن كل نظام تشغيل يتعامل مع أقسام سجل الإقلاع بأسلوب مختلف قليلا ".
5. ^ أ ب، نورتن، بيتر ؛ كلارك، سكوت Norton, Peter; Clark, Scott (2002). كتاب "*Peter Norton's New Inside the PC*". الناشر Sams Publishing. صفحات 360-361 . ISBN 0-672-32289-7.
6. ^ مايكل جريفز Graves, Michael W (2004). كتاب "*A+ Guide To PC Hardware Maintenance and Repair*". الناشر Thomson Delmar. صفحة 276.. ISBN 1-4018-5230-0 .
7. ^ أندروز جين Andrews, Jean (2003). كتاب "*Upgrade and Repair with Jean Andrews*". الناشر Thomson Course Technology. صفحة 646 . ISBN 1-59200-112-2".
8. ^ بوزويل وليام Boswell, William (2003). كتاب "*Inside Windows Server 2003*", الناشر Addison-Wesley Professional. صفحة 13. ISBN 0-7357-1158-5.
9. ^ سميث رودريك Smith, Roderick W (2000). كتاب "*The Multi-Boot Configuration Handbook*". الناشر Que Publishing. صفحات 260-261 . ISBN 0-7897-2283-6.
10. ^ أندريس إيفرت بروير، Brouwer, Andries Evert "خصائص جداول الأقسام" صفحة أنواع الأقسام. اقتباس عن ماتياس بول: "نظام تشغيل PTS-DOS [يستخدم] مدخلة خامسة خاصة للقسم مقابل المدخلات الأربعة الأخرى في MBR، ترتبط بشفرة إقلاع تفهم أقسام AAP."
11. ^ أندريس إيفرت بروير، "خصائص جداول الأقسام"، صفحة أنواع الأقسام. "اقتباس عن ماتياس بول Matthias Paul: "بعض أنظمة صانعي القطع الأصلية OEM، مثل AST DOS (النوع 14h) و NEC DOS (النوع 24h) تملك 8 مدخلات للأقسام بدل 4 في قطاع MBR". (ملحوظة: جداول أقسام 8 مدخلات في MS-DOS 3.30 و NEC و AST مسبوقة بتوقيع A55Ah الحيد +17Ch). 8 مدخلات أولية، أولاها تبدأ عند الحيد 07EEh وتمتد إلى الحيد 07FDh، المدخلة الثانية ستكون فوق الأولى مباشرة. أي بترتيب خلفي، عكس المعمول به في MBR المعياري. لكن برنامج NEC FDISK يستطيع أيضا إنشاء 4 مدخلات أولية فقط (معيارية) للتوافق مع أنظمة دوس الأخرى.

الشفرة باللون الأزرق موجودة فقط في NEC MS-DOS 3.30 (هذه نسخة معدلة من MS-DOS 3.30 MBR)

```
      00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F  0123456789ABCDEF
7C00  FA 33 C0 8E D0 BC 00 7C 8B F4 50 07 50 1F FB FC |.3.....|..P.P...|
7C10  BF 00 06 B9 00 01 F2 A5 EA 1D 06 00 00 BE BE 07 |................|
7C20  B3 04 BF 7C 07 81 3D 5A A5 75 05 B3 08 BE 7E 07 |...|..=Z.u....~.|
7C30  80 3C 80 74 0E 80 3C 00 75 1C 83 C6 10 FE CB 75 |.<.t..<.u......u|
```

موقع جديد للشفرة. 0600. ثم قفزة إلى 061D

```
061D
```

بقية الشفرة حتى بداية التوقيع .....الخ

جدول أقسام إن إي سي مايكروسوفت دوس 3.30 في MBR (عناوين في الذاكرة)

```
0770                                  5A A5 00 00 |              Z...|
0780  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
0790  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
07A0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
07B0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
07C0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
07D0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
07E0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 80 01 |................|
07F0  01 00 24 0F FF 82 3F 00 00 00 91 D3 0D 00 55 AA |..$...?.......U.|
      00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F  0123456789ABCDEF
```

نفس الشفرة مع. نفس الحيود بداية من 7C00.. .. الخ

ثم قفزة إلى نسخة جديد من الشفرة..

```
7C00 FA. CLI
. . . . .etc
7C18 EA1D060000  JMP  0000:061D
```

العناوين التالية تعكس الموقع الفعلي للشفرة في الذاكرة زمن التنفيذ. لأن الروتين السابق قبل أن يستمر، ينسخ نفسه مع كامل الشفرة التي تليه إلى الموقع 0600 ثم يقفز إلى 061D .

```
061D BEBE07  MOV  SI,07BE
0620  B304  B304 MOV  BL,04
```

هذه السطور توجد فقط في: NEC MS-DOS 3.30 MBR
(نسخة معدلة من: MS-DOS 3.30 MBR)

```
0622  BF7C07  MOV  DI,077C
0625  813D5AA5  CMP  WORD PTR [DI],A55A
0629  7505  JNZ  0630
062B  B308  MOV  BL,08
062D  BE7E07  MOV  SI,077E
```

التحقق من القسم القابل للإقلاع.
(نفس الشفرة مع. اختلاف الحيود فقط). . . . الخ
أي (0630 بدل 0622 الذي في MS-DOS 3.30 MBR).

```
0630  803C80  CMP  BYTE PTR [SI],80
. . . .etc
```

هذه الشفرة تتفحص موقع الذاكرة **077C**. (أنظر أعلاه).إذا لم تجد الكلمة الست عشرية A55Ah، تقفز إلى الموقع **0630** وتكمل بقية الشفرة كما تفعل الشفرة المعيارية MS-DOS 3.30. لكن إذا وجدت ذلك التوقيع، القيم الموضوعة سابقا في التسجلان **BL** و **SI**. (04 و 07BE) تستبدل بقيم بديلة من **08** و **077E**. هذا يدل بوضوح، على وجود 8 مدخلات أولية في جدول الأقسام، تبدأ من الحيد **077E** (بدل 4 مدخلات عادية تبدأ عند الحيد **07BE**). هذه المدخلات يجب تنصيبها هناك باستخدام نفس البرنامج الذي يعين بايتات **5A** و **A5** في الموقع 077C و 077D (أي باستخدام برنامج NEC FDISK) والتي ينتج عنها جدول أقسام MBR بحجم يشبه هذا (أنظر أعلاه).

تنبيه: تم حذف تدوين القطعة:0000 من عناوين الذاكرة في جميع الأمثلة (طرح الشفرة) الواردة في هذه الكتيبات: مثال عنوان الذاكرة **7C00**:0000 يظهر فقط **7C00**.

12. ^ سيدوري دانيال , "ملاحظات عن اختلافات إحدى نسخ OEM في MBR - نظام دوس 3.30". صفحة *Master Boot Records*. اقتباس: "عندما أضفنا أقسام إلى جدول NEC، وقعت مدخلة القسم الأول من الحيود 1EEh+ حتى 1FDh+ والمدخلة التالية كانت فوقها مباشرة. أي أن، المدخلات كانت مقحمة ومرتبة بشكل عكسي يخالف ما هو معروف في الجدول العادي. وبالتالي، تفحص مثل هذا الجدول باستخدام محرر للقرص أو وسيلة لعرض الأقسام، سيعرض المدخلة الأولى التي في جدول المدخلات الثمانية NEC كأخر مدخلة (أي المدخلة الرابعة) في جدول الأقسام العادي".عرض جدول أقسام 8-مدخلات وأين تختلف شفرة إقلاعها عن MS-DOS 3.30.

13. ^ "جدول الأقسام" موقع osdev.org . جدد في 15-11-2013.

14. ^ أ ب ت , كتاب "*System BIOS for IBM PC/XT/AT Computers and Compatibles*" مرجع فينكس الفني. إديسون وزلي. 1989. رقم ISBN 0-201-51806-6.

15. ^ أندريس إيفرت بروير، "لائحة معرفات الأقسام في الحاسوب الشخصي" ، صفحة أنواع الأقسام.

16. ^ سيبيل وود Wood, Sybil (2002). كتاب "*Microsoft Windows 2000 Server Operations Guide*". صفحة 18. الناشر Microsoft Press رقم ISBN 9780735617964.

17. ^ "مقدمة في هندسة القرص الثابت". الناشر *Tech Juice*. تاريخ 08-08-2011. جدد في 19-04-2013.

18. ^ تشارلز كوزيروك Kozierok, Charles M. (17-04-2001). "نظام BIOS والقرص الثابت". موقع "*The PC Guide*". جدد في 19-04-2013 .

19. ^ سميث روبرت Smith, Robert (26-06-2011). "تجاوز قيود سجل الإقلاع الرئيسي". الدروس الخاصة"*GPT fdisk Tutorial*. جدد في 20-04-2013.

20. ^ "أكثر من 2 ترابايت على قرص MBR". موقع superuser.com . تاريخ 07-03-2013 . جدد في 22-10-2013.

21. ^ "التحول إلى إستخدام أقراص صيغة/التهيئة المتقدمة، قطاع 4 كيلوبايت". صفحة *Tech Insight*. شركة Seagate Technology. جدد في 19-04-2013.

22. ^ كيلفن كالفرت Calvert, Kelvin (16-03-2011). ملف (PDF) "الأقراص الثابتة ذات السعة الكبيرة WD AV-GP". شركة Western Digital. جدد في 20-04-2013 .

23. ^ سميث رودريك (27-04-2010). "نظام تشغيل لينكس على أقراص قطاع 4-كيلوبايت : نصيحة". موقع IBM DeveloperWorks. جدد في 19-04-2013.

24. ^ أ ب، "سجل الاقلاع الرئيسي (x86)". موقع OSDev.org . *OSDev Wiki*. جدد في 20-04-2013.

25. ^ سيدوري دانيال (30-07-2003). "سجل الإقلاع الرئيسي في نظام تشغيل IBM DOS 2.00". جدد في 22-07-2011.

26. ^ سينغ أميت Singh, Amit (25-12-2009). "إقلاع نظام تشغيل ماك عشرة Mac OS X". جدد في 22-07-011.

27. ^ جوناثان دي بوين بولارد de Boyne Pollard, Jonathan (10-07-2011). "عملية إقلاع EFI". صفحة "الأسئلة و الأجوبة المكررة". جدد في 22-07-2011.

28. ^ دومسك مات Domsch, Matt . "رد: تحسينات القرص، معيار RFC 2.6.0 EDD". قائمة بريد نواة لينكس.

29. ^ "نظام ويندوز قد يستخدم الصاغة (Signature) في ملف BOOT.INI". صفحات *KnowledgeBase* . مايكروسوفت.

30. ^ "توقيع القرص في سجل الإقلاع الرئيسي - نظام تشغيل ويندوز فيستا". "تشغيل وإقلاع مزدوج مع ويندوز فيستا". يناير 2007 جدد في 19-04-2013.

31. ^ مارك روسينوفيش Russinovich, Mark (08-11-2011). "إصلاح تعارض تواقيع القرص". مدونة:Mark Russinovich . مايكروسوفت. جدد في 19-04-2013.

32. ^ أ ب ت، ساكاموتو ماساهيكو Sakamoto, Masahiko (13-05-2010). "لماذا نظام BIOS يحمل سجل MBR عند العنوان 0x7C00 في نظام x86 ؟". جدد في 04-05-2011.

33. ^ أ ب ت ث ج ح، كومباك؛ فينكس؛ إنتيل (01-11-1996). ملف (PDF) "مواصفة إقلاع نظام BIOS رقم 1.01". ACPI-CA. جدد في 20-04-2013.

34. ^ أ ب ، إليوت ديفيد Elliott, David (12-10-2009). موضوع "لماذا يقوم سجل الإقلاع "المعياري" بتعيين التسجيل SI ؟". جدد في 20-04-2013.

35. ^ أ ب ت، كومباك؛ فينكس؛ إنتيل (05-05-1994). ملف (PDF) "مواصفة نظام BIOS الذي يدعم معيار "القس والتشغيل - رقم 1.0A". إنتيل. جدد في 20-04-2013.

36. ^ إليوت روبرت Elliott, Robert (04-01-2010). "ملحق شفرة إقلاع MBR الهجين مع مواصفة الأقراص، نسخة EDD-4". موقع لجنة المعايير t13.org. جدد في 20-04-2013.

37. ^ "الأمر FDISK /MBR بعد كتابة MBR". صفحة الدعم. موقع مايكروسوفت. تاريخ 23-09-2011. جدد في 19-04-2013.

38. ^ "صفحة المساعدة – نظام لينكس – برنامج sfdisk (8)". جدد في 20-04-2013.

39. ^ رالف براون Brown, Ralf (16-07-2000)، "لائحة مقاطعات رالف براون" (v61 html)"، برمجيات *Delorie Software*، مشروع *DJGPP* من أجل دوس في تاريخ 03-11-2016

40. ^ رالف براون (16-07-2000)، "لائحة مقاطعات x86 (النص الأصلي v61) في ملف الأرشيف "*inter61a.zip*": من"INTERRUP.B", العنوان: "B-1302" أو "INT 13 - DISK - READ - SECTOR(S) INTO MEMORY"". لائحة مقاطعات رالف براون (*RBIL*) في تاريخ 03-11-2016.

41. ^ ياكوب كامينسكي Jakub kaminsky، نشرة الفيروسات "Rainbow: To Envy or to Hate" سبتمبر/أيلول 1995، الصفحات 2-7. موقع computervirus. جدد في تاريخ 3-2017..

42. ^ مايك لامبرت Mike Lambert، نشرة "Circular Extended Partitions: Round and Round with DOS"، سبتمبر/أيلول 1995، الصفحة 14. موقع computervirus. جدد في تاريخ 3-2017..

43. ^ وثائق ومواصفة UEFI.

تنبيه

لا توجد أية مصادر عربية في هذا الكتيب! باستثناء بعض المصطلحات القليلة من قاموس عرب آنز. وبعض الفقرات من الموسوعة الحرة – العربية.

احتمال وجود أخطاء في هذا الكتيب وارد. وسواء كان الخطأ من المصدر الانجليزي أو من الترجمة العربية. إذا كنت متخصص أو مدون يمكنك مراجعة ومقارنة الكتيب بالمصدر الانجليزي للترجمة. وتصحيحها في كتابتكم مع الإشارة إلى المصدر أو تصحيحها وإرسالها بالبريد الإلكتروني أو على المدونة.

تنبيه: النسخة "المراجعة" لا تعني بالضرورة عدم وجود أخطاء.... فالكامل لله وحده.

جهاد

مراجعة في يناير/كانون الثاني 2019

تمت بحمد الله