سجل إقلاع القسم

VOLUME BOOT RECORD

مسوّدة

يناير/كانون الثّاني 2019

سجل إقلاع وحدة التخزين

يسمى كذلك :سجل إقلاع القسم PBR، قطاع إقلاع القسم، قطاع إقلاع وحدة التخزين.

في أقراص BIOS/MBR، سجل VBR جزء من منطقة القطاعات المحجوزة في بداية وحدة التخزين. عادة يدعى قطاع إقلاع (كتلة إقلاع) خصوصا إذا كان بطول قطاع واحد (غالبا 512 بايت) أو سجل إقلاع إذا كان بطول عدة قطاعات كما هو الحال مع سجلات إقلاع وحدة التخزين FAT32 و NTFS في أنظمة مايكروسوفت دوس/ويندوز.

سجل الاقلاع موجود في أجهزة تخزين البيانات المتوافقة مع الحاسوب الشخصي، المقسمة مثل القرص الثابت، والقابلة للإزالة مثل ذاكرة الفلاش وقرص Zip، وفي الغير مقسمة مثل القرص المرن. هذا القطاع ينشأ عند تهيئة القسم (أي وحدة التخزين) تهيئة منطقية أو ما يعرف بتهيئة المستوي العالي مثلا، بعد تقسيم القرص واستخدام الأمر FORMAT في دوس/ويندوز، أو مدير القرص في ويندوز أن تي أو برنامج GParted في إحدى توزيعات لينكس (القرص المدمج الحي).. إلى آخره.

في أنظمة ملفات، مثل HPFS ،NTFS، و FAT (باستثناء دوس DOS 1.x)، قطاع الاقلاع يتضمن بنية للبيانات BPB (تصف موقع وتخطيط بنية البيانات على الوسيط) وشفرة برنامج إقلاع، مكتوبة بلغة الآلة (ليس بالضرورة، لكن قد تكون جزء من نظام تشغيل) مخزن في جزء أخر من القرص.

في الأجهزة الغير مقسمة قطاع الاقلاع هو أول قطاع على القرص. وفي الأجهزة المقسمة هو أول قطاع في القسم الأولى أو القرص المنطقي داخل القسم الممتد على القرص الثابت، (تحديدا، القطاع المنطقي 0 في القسم الأول عند العنوان الفيزيائي C0/H1/S1 وفي الأقسام اللاحقة في القطاع الأول زائد إزاحة القسم) علما أن القطاع الأول في كامل القرص الثابت سيكون سجل MBR الذي يتضمن جدول أقسام ومعلومات أخرى.

VBR أقل معيارية من MBR الذي يتضمن أيضا شفرة وبيانات، لكن VBR مثل MBR شفاف للنظام التشغيل؛ موجود خارج منطقة بيانات القرص التي عليها تخزين الملفات. كما قلت، سجل إقلاع القسم يحتفظ ببعض المعلومات من أجل نظام الملفات، مثل، معاملات BPB، (رغم أنها ليست مطلوبة في الأنظمة الحديثة لكن العديد من أنظمة التشغيل تحتفظ بها للتوافق) وشفرة الاقلاع التي تحددها دائما أنظمة التشغيل، لكنها تعمل جميعها تقريبا بنفس الطريقة؛ أي بتحديد موقع النواة على القسم، ثم تحميلها وتنفيذها. أو تحميل ملف برنامج البدء كما يفعل قطاع إقلاع نظام دوس، بالتعرف على نظام الملفات FAT وتحميل ملفات محمل الاقلاع المرحلة الثانية IO.SYS و MSDOS.SYS * (أي ملفات التهيئة والمشغلات والنواة) من الدليل الجذر.

> * MSDOS.SYS ملف مهم في مايكروسوفت دوس MS-DOS و ويندوز Windows 9x. الملف يحمل من قبل IO.SYS. في أنظمة MS-DOS، هذا الملف يتضمن الشفرة الأساسية لنظام التشغيل، أي نواة دوس. أما ملف IO.SYS. فهو عنصر أساسي في أنظمة MS-DOS و Windows 9x. يتضمن مشغلات الأجهزة (مشغلات العتاد) في نظام MS-DOS (أي روتينات ربط العتاد) وبرنامج تهيئة (تمهيد) دوس.

طول ومحتوى سجل إقلاع القسم يتفاوت بحسب نوع نظام الملفات المستخدم. لكنها جميعا تشترك في بعض الخصائص. سجل VBR يحتوي على عدة عناصر مشابها لعناصر MBR، مثل شفرة الاقلاع، وبيانات القرص، وتوقيع الاقلاع 55AAh . وتقريبا كل سجل إقلاع قسم في أنظمة مايكروسوفت كان وما زال يتضمن العناصر المحددة التالية:

بنية قطاع الاقلاع			طول (بايت)			إزاحة		
اسم	رمز تذكري	وظيفة	FAT16	FAT32	NTFS	FAT16	FAT32	NTFS
تعليمة القفزة	JMP	من أجل تجاوز منطقة البيانات	3	3	3	000h (0)	000h (0)	000h (0)
هوية صانعي القطع الأصلية	OEM ID	منطقة البيانات (الخصائص الفيزيائية للوسيط)	8	8	8	003h (3)	003h (3)	003h (3)
معاملات الكتلة المعيارية	BPB		25	53	25	00Bh (11)	00Bh (11)	00Bh (11)
معاملات الكتلة الممتدة	EBPB		26	26	48	024h (36)	040h (64)	024h (36)
شفرة إقلاع	IPL [14]	شفرة تتبع محمل الاقلاع	448	420	426	03Eh (62)	05Ah (90)	054h (84)
توقيع القطاع	55AAh	تدل على قطاع تنفيذي (يقبل الاقلاع)	2	2	2	1FEh (510)	1FEh (510)	1FEh (510)

نوع قطاع الاقلاع	عدد القطاعات		نسخة احتياطية	أنظمة التشغيل
	المحجوزة	المستخدمة		
FAT12 / FAT16	1	1	لا توجد	دوس، جميع إصدارات مايكروسوفت
FAT32	32	3 من 6	في المنطقة المحجوزة	دوس، مايكروسوفت ويندوز
NTFS	16	6 + 1	في أخر قطاع في القسم	مايكروسوفت ويندوز

تعليمة القفزة / التفرع "اللامشروطة"

في برمجة الحاسوب، هذا التعبير يشير إلى تعليمة "القفز من مكان إلى آخر في الشفرة" مع "تجاوز البيانات أو الشفرة التي لا ينبغي تنفيذها." أي تغير تدفق البرنامج/النظام control flow. وتقريبا، جميع زمر التعليمات في لغة الآلة؛ اللغة المخفضة المستوى، تتضمن هذه التعليمة التي تدعى أيضا تعليمة تفرع Branch أو [18] قفزة jump في أنظمة x86 ويشار لها عادة بالرمز التذكري JMP. أو BRA (كما في موتورولا MC68K). وتدعى بأسماء أخرى مثل GO TO، goto في لغات البرمجة عالية المستوى.

كما ذكرنا، تعليمة القفزة أو التفرع اللامشروطة، تغير تدفق البرنامج control flow أي تغير متتالية تنفيذ التعليمات في البرنامج، دون تسجيل معلومات العودة، وهي تحويل التنفيذ إلى سطر آخر من شفرة البرنامج دون وجود شرط يحتاج إلى الفحص (كما في "اللامشروطة"؛ صحيح كان أو خاطئ. عنوان التعليمة التي يتم القفز إليها يحدده معامل الوجهة (الهدف). المعامل يمكن أن يكون:

• قيمة لحظية (فورية)	• immediate value
• تسجيل غرض عام	• general-purpose register
• موقع ذاكرة	• memory location

القفزة اللامشروطة ثلاثة أنواع في ISA أو مجموعة تعليمات معالج ميكرو (أي الأوامر التي ينفذها المعالج)

- قفزة بعيدة 5 بايت تسمح بالقفز إلى أي موقع ذاكرة (بدلا من قطعة الشفرة الحالية لكن على نفس مستوى الامتياز privilege level) ضمن نظام الذاكرة الحقيقي. يستخدم الأمر التوجيهي FAR directive لإعلام المجمع باستخدامها.
- قفزة القطعة البينية المباشرة Inter segment Direct Jump، تدعى أيضا بقفزة بعيدة مباشرة Far Direct Jump، وهي برنامج قفزة قطعة بينية شائع. طول التعليمة : 5 بايت. 1 بايت شفرة تشغيل EAh، و 2 بايت قيمة الإزاحة، و 2 بايت قيمة القطعة.
- قفزة القطعة البينية الغير مباشرة Inter segment Indirect Jump، تدعى أيضا بقفزة بعيدة غير مباشرة Far Indirect Jump، وهي نادرة الاستخدام. طول التعلمية يعتمد على طريقة تحديد موقع القفزة. يمكن أن تكون 2 بايت كحد أدنى.

- قفزة قصيرة 2 بايت تسمح بالقفز / التفرع إلى موقع ذاكرة ضمن مدى: -128 إلى +127 بايت. من العنوان الذي يتبع القفزة (القيمة الحالية في EIP/IP). يمكن أن يستخدم الأمر التوجيهي SHORT directive لإعلام المجمع باستخدامها.
- قفزة قريبة 3 بايت تسمح بالقفز / التفرع ضمن إزاحة من ±32 كيلوبايت من التعليمة في قطعة الشفرة الحالية. يستخدم الأمر التوجيهي NEAR directive لإعلام المجمع باستخدامها.
- تعليمة القفزة الطويلة Long Jump، تعني : تعليمة 3-بايت. المدى من 32768- إلى 32767+ بايت. القفزة الطويلة يمكن أن تغطي كامل 64 كيلوبايت من قطعة الشفرة CS.
- قفزة القطعة الداخلية الغير مباشرة Intra segment indirect Jump، تدعى أيضا بقفزة قريبة غير مباشرة Near Indirect Jump، لا تستخدم كثيرا (نادرة). طول التعليمة : 2 بايت أو أكثر. المدى: قطعة كاملة

قفزات القطعة الداخلية الثلاثة (المطلقة والنسبية) والقطعة البينية الاثنان (المطلقة) تملك نفس الرمز التذكري JMP، لكنها تختلف في شفرة التشغيل.

البعض يدعو القفزة البعيدة بالقفزة الطويلة Long Jump. ويدعو القفزة القصيرة والقريبة بالقفزة القصيرة Short Jump.

تصنف أيضا القفزة اللامشروطة إلى أربعة أنواع بإضافة القفزة القريبة الغير مباشرة FF، أي EB, E9, EA, FF. وتصنف إلى أربعة أنواع كذلك بإضافة قفزة Task switch وتعني قفزة إلى تعليمة تقع في مهمة (أخرى) مختلفة، تستخدم فقط في النمط المحمي للمعالج.

وصف	تعليمة	شفرة تشغيل
قفزة قصيرة، نسبية، الإزاحة مرتبطة بالتعليمة التالية	JMP rel/8	EB cb
قفزة قريبة، نسبية، الإزاحة مرتبطة بالتعليمة التالية	JMP rel/16	E9 cw
قفزة قريبة، نسبية، الإزاحة مرتبطة بالتعليمة التالية	JMP rel/32	E9 cd
قفزة قريبة، مطلقة غير مباشرة. [33]، العنوان معطى في r/m16	JMP r/m16	FF /4
قفزة قريبة، مطلقة غير مباشرة، العنوان معطى في r/m32	JMP r/m32	FF /4
قفزة بعيدة، مطلقة، العنوان معطى في المعامل	JMP ptr16:16	EA cd
قفزة بعيدة، مطلقة، العنوان معطى في المعامل	JMP ptr16:32	EA cp
قفزة بعيدة، مطلقة غير مباشرة، العنوان معطى في m16:16	JMP m16:16	FF /5
قفزة بعيدة، مطلقة غير مباشرة، العنوان معطى في m16:32	JMP m16:32	FF /5

rel8	> عنوان إزاحة نسبية 8-بت. (المدى من 128 بايت قبل إلى 127 بايت بعد نهاية التعليمة)			
r/m8	r/m16	r/m32	> تسجيل غرض عام (تسجيل بيانات) أو موقع ذاكرة (8-بت و 16-بت و 32-بت)	
ptr16:16	ptr32:32	> مؤشر بعيد مزدوج 16-بت أو 32-بت في قطعة شفرة مختلفة		
m16:16	> موقع ذاكرة يتضمن مؤشر بعيد مركب من عدد مزدوج : قطعة 16-بت و إزاحة 16-بت			
rel16	rel32	> عنوان إزاحة نسبية 16-بت أو 32-بت ضمن نفس قطعة الشفرة		
m16:32	> موقع ذاكرة يتضمن مؤشر بعيد مركب من أعداد: قطعة 16-بت وإزاحة 32-بت			

> 1-بايت (cb) ، 2-بايت (cw) ، 4-بايت (cd) ، 6-بايت (cp) ، 8-بايت (co)، أو 10-بايت (ct) قيمة تتبع المعامل. هذه القيمة تستخدم في تحديد إزاحة الشفرة وربما قيمة جديد في تسجيل قطعة الشفرة CS.

> FF/x = رقم ما بين 0 و 7 يشير إلى أن بايت التعليمة في ModR/M يستخدم فقط معامل التسجيل أو الذاكرة r/m . حقل التسجيل يتضمن الرقم الذي يوفر امتداد إلى شفرة تشغيل التعليمة. (لمعلومات أكثر راجع دليل Intel)

تعليمة القفزة القصيرة Short Jump

القفزة القصيرة تدعى قفزة نسبية relative jump لإمكانية ترحيلها إلى أي مكان ضمن قطعة الشفرة الحالية دون تغيير. ولأن عنوان القفزة لا يخزن مع شفرة التشغيل. هذه الأخيرة تتبعها مسافة أو إزاحة displacement. بدلا من عنوان قفزة. إزاحة القفزة القصيرة "مسافة" يمثلها عدد مؤشر 1-بايت، يكون موجب مع إشارة للأمام وسالب مع إشارة للخلف، المدى: -128 إلى +127 بايت.

هذه الإزاحة تضاف إلى عنوان التعليمة التالية من أجل إيجاد عنوان الهدف.

أنوع القفزات المشروطة (مثل، JE, JG, JC, JZ, JNE, JNG, JNC, JNZ, إلخ) تعرف أيضا بالقفزات النسبية القصيرة SHORT Relative Jumps. البرامج التي تستخدم فقط تعليمات القفزة النسبية يمكنها إعادة التموضع في أي مكان في الذاكرة دون الحاجة إلى تغيير لغة الآلة من أجل القفزات.

أول بايت من القفزة القصيرة (اللامشروطة) SHORT Jump دائما EBh والبايت الثاني إزاحة نسبية relative offset من الحيد 00h إلى 7Fh للقفزات إلى الأمام Forward jumps، ومن 80h إلى FFh للقفزات إلى الخلف (أو المعكوسة) Backward jumps. تعداد الإزاحة يبدأ دائما عند البايت مباشرة بعد تعليمة JMP مع أي نوع من القفزات النسبية !.

إزاحة	شفرة التشغيل
-128 إلى +127 بايت	EB

قفزة قصيرة مع مدى : -128 إلى +127 بايت. يتأثر فقط تسجيل IP

عندما ينفذ المعالج قفزة قصيرة، إشارة الإزاحة تتمدد وتضاف إلى مؤشر التعليمة IP/EIP (يدعى أيضا : عداد برنامج) لتولد عنوان القفزة ضمن قطعة الشفرة الحالية. تعليمة القفزة القصيرة تتفرع إلى هذا العنوان الجديد للتعليمة التالية في البرنامج (IP/EIP يتضمن عنوان التعليمة التالية التي ستنفذ). أنظر للشكل أدناه.

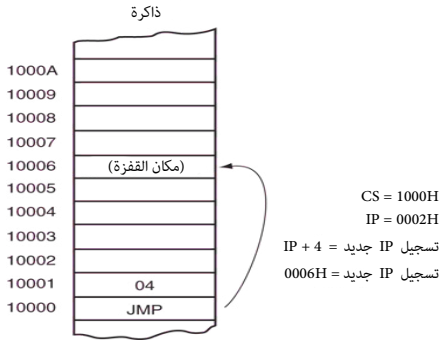
عنوان القفزة يمكن أن يحدد عن طريق معرف label [10] مثلا JMP NEXT، تقفز إلى NEXT للتعليمة التالية. نحن لا نستخدم أبدا هنا عنوان ست عشري فعلي مع أي تعليمة قفزة. المعرف يتبعه نقطتان (:NEXT) كي تتم القفزة.إذا لم تتبعه نقطتان، لا يمكن القفز إليه. وتستخدم النقطتان فقط عند استخدم المعرف مع تعليمة قفزة jmp أو نداء call. القفزات اللامشروطة إلى labels هي قفزات نسبية relative jumps. مثال.

```
0000 EB 03   JMP STOP
0002 90      NOP
0003 33 00   XOR AX, AX
0005 B4 4C   STOP:  MOV AH, 4Ch
```

الإزاحة = الاختلاف بين معرف الهدف وIP (يمكن كذلك أن تكون سالبة)

ينفذ المعالج القفزة بإضافة الإزاحة إلى قيمة IP الحالية (IP := 0002 + 3 = 0005) => IP سيشير إلى التعليمة التي عندها سوف يستمر تنفيذ البرنامج.

وسواء استخدمت معرف label للإشارة إلى التعليمة التالية أو استخدمت عنوان مخصص (كما يطلب أمر Assemble في برنامج مثل Debug)، كافة المجمعات ستظل قادرة على كشف قيمة بايت الإزاحة. إذا أشرت إلى عنوان بعيد جدا عن متناول القفزة القصيرة Short jump، المجمع سيحول التعليمة إلى قفزة 3 بايت قريبة Near jump *. (علما أن القفزة المطلقة البعيدة Absolute FAR Jump هي قفزة خارج قطعة الشفرة CS الحالية 64 كيلوبايت) ولهذا، المبرمج الذي يحاول إبقاء الروتين بأقل عدد من بايتات، يجب أن يعرف حدود كلتا القفزتين القصيرتين إلى الأمام والخلف (والقفزة القريبة).



قفزة قصيرة إلى عنوان التعليمة التالية. التسجيل IP + إزاحة هدف ممتد مع إشارة -> IP

* المجمع (أو أمر Assemble) سوف يستخدم أصغر شفرة JMP ممكنة مع أي عنوان يقدم له (أولا، القصيرة SHORT، ثم القريبة NEAR، وأخيرا البعيدة FAR). السبب في قدرته على فعل ذلك لأن موقع التعليمة التالية بالضبط سيكون محدد. رغم ذلك، معظم المجمعات، سوف تنشئ حيز على الأقل من أجل قفزة 3 بايت القريبة Near jump حتى وإن كانت غير ضرورية ؛ ما لم تضمن أنت الأمر التوجيهي directive "SHORT" للقفزة القصيرة قبل الرمز التذكري "JMP" في الشفرة الأصلية ! قد يفسر هذا لماذا تشاهد تعليمة لا عملية NOP أي البايت 90h بعد القفزة القصيرة SHORT Jump في الشفرة التي لا تحتاج إلى بايت إضافي. مع استعمال فقط اسم المعرف label، المجمعات تحتاج أكثر من مرور PASS واحد من خلال الشفرة الأصلية لمعرفة كم يبعد (من تعليمة القفزة) اسم ذلك المعرف المشار إليه فعليا. إذا استخدمت في شفرتك الأصلية، أمر توجيهي للقصيرة SHORT directive وكان العنوان بعيدا جدا عن متناول القفزة القصيرة، فسوف تحصل على رسالة خطأ.

قفزة إلى الأمام JMP FRWD

المجمع لا يعرف مقدار القفزة في المرور pass.1. المجمع يحجز 3 بايت لتعليمة القفزة إلى الأمام. إذا ثبت أن مسافة القفزة > 128 بايت، ترمز التعليمة إلى E9 r16 (و E9h = شفرة قفزة طويلة). إذا مسافة القفزة أصبحت <= 128 بايت، ترمز التعليمة إلى EB r8 متبوعة بشفرة لا عملية NOP (و EBh = شفرة قفزة قصيرة).

القفزات إلى الأمام أسهل في التعامل مقارنة بالقفزات إلى الخلف، فهي تستخدم قيم للإزاحة النسبية من 00h إلى 7Fh تمكن تنفيذ البرنامج القفز إلى تعليمة أخرى بينها كحدى أقصى 127 بايت. أما بايت الإزاحة النسبية، فهو أساسا، عدد مؤشر 8-بت حيث البت أو الخانة الأكثر أهمية MSB هي **0** للأعداد الايجابية. ولهذا، كافة بايتات من **0** وحتى **7Fh** (في الثنائي **0**111 1111) هي إيجابية وتعطينا قفزة إلى الأمام Forward Jump.

قفزة إلى الخلف JMP BKWD

المجمع يعرف مقدار القفزة ويولد شفرة القفزة القصيرة Short Jump إذا القفزة المطلوبة <= 128 بايت. ويولد شفرة القفزة الطويلة long Jump إذا القفزة المطلوبة > 128 بايت.

القفزات إلى الخلف تملك بايتات إزاحة نسبية من 80h إلى FFh . خلافا للقفزات إلى الأمام، بايت الإزاحة الذي يبدوا الأكبر هنا في الواقع يشير إلى أقصر قفزة إلى الخلف، لأننا يجب أن نستخدم المتمم الثنائي 2 * من كل بايت إزاحة مع إشارة سالبة، دعنا نقوم بحساب المتمم الثنائي 2 لكلا الحدين الأعلى والأدنى من القفزة القصيرة إلى الخلف:

أولا، نعكس كل بت من بايت الإزاحة (الذي يعطي متممه الأحادي 1):

80h (1000 0000) —> **7Fh** (**0**111 1111) و **FFh** (1111 1111) —> **00h** (**0**000 0000)

بإتباع هذا، ببساطة تضيف 1 إلى كل قيمة وسيطة، ثم تحولها إلى عدد سالب. إذن، المتمم الثنائي 2 من كل بايت هو في الواقع:

80h —> **-80h** (-128) و **FFh** —> **-01h** (-1)

هذه الأعداد السالبة أيضا إلكترونية، وإلا لن تكون هناك قفزات إلى الخلف (المعالج يعلم أنها حيود سالبة لأن أول بايت EBh، يخبره أنها تعليمة Short Jump حيث أية قيمة من 80h إلى FFh تعامل بهذه الطريقة).

> * إذا استخدمنا فقط الأعداد المؤشرة 8-بت البسيطة، إذن 00h سوف يعطينا 0+ (صفر موجب؛ في الحقيقة، كما تعلم، الصفر محايد بين مجموعتي الأعداد فلا هو سالب ولا هو موجب ويوضع بالمنتصف بين المجموعتين)، لكن 80h (في الثنائي 0000 1000) سوف يعطينا 0- (صفر سالب)! وهذا، المبرر الأول لاستخدام حساب المتمم الثنائي 2 من أجل تجنب الحصول على صفران مختلفان ! (أي وجود قيميتين للصفر).

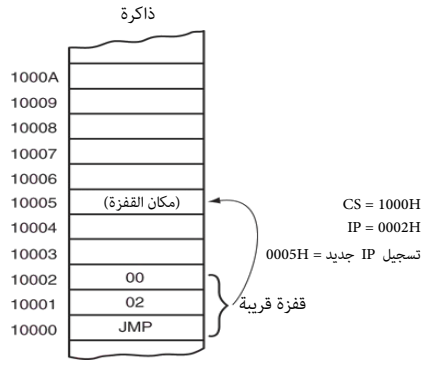
أمر المجمع التوجيهي للقصيرة SHORT Assembler Directive.

- المجمع يولد فقط شفرة قفزة قصيرة 2 بايت من أجل القفزة إلى الأمام، إذا استخدم الأمر التوجيهي للمجمع للشفرة القصيرة.

تعليمة القفزة القريبة Near Jump

القفزة القريبة تشبه القصيرة (ليس هناك اختلاف بين ترميز القفزة القصيرة النسبية والقريبة النسبية) باستثناء أن مسافة القفزة أبعد. القفزة القريبة تمرر التحكم إلى التعليمة في قطعة الشفرة الحالية الواقعة ضمن ± 32 كيلوبايت من تعليمة القفزة القريبة. المجمع يستخدم القفزة القصيرة إذا كان الهدف ضمن المدى القصير حتى يولد شفرة موجزة أكثر. ويستخدم القفزة القريبة بشكل آلي إذا كان الهدف يبعد أكثر من 128 بايت.

تعليمة القفزة القريبة 3-بايت تتضمن شفرة تشغيل يتبعها رقم إزاحة 16-بت مع إشارة. الإزاحة ذات الإشارة تنضم إلى مؤشر التعليمة IP لتولد عنوان القفزة. لأن مدى الإزاحة ذات الإشارة هو ±32 كيلوبايت، يمكن للقفزة القريبة القفز إلى أي موقع ذاكرة ضمن قطعة الشفرة الحالية في النمط الحقيقي.



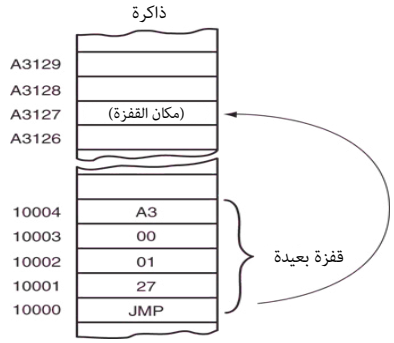
قفزة قريبة تضيف الإزاحة (0002H) إلى محتويات التسجيل IP

إزاحة	إزاحة	شفرة التشغيل
بايت عالي	بايت منخفض	E9

قفزة قريبة ضمن نفس القطعة (-32,768 إلى+32,767). يتأثر فقط تسجيل IP . (التسجيل IP + (إزاحة هدف)) -> IP . أو قفزة غير مباشرة ؛ (تسجيل أو قيمة في الذكرة) -> IP

تعليمة القفزة البعيدة Far Jump

القفزة البعيد حتى تنجز القفزة، تحصل على عنوان قطعة وإزاحة جديدين. البايت 2 و 3 في تعليمة 5-بايت تتضمن عنوان الإزاحة الجديد والبايت 4 و 5 تتضمن عنوان القطعة الجديد، وفي كلتا الحالتين البايت العالي يتبع البايت المنخفض.



تعليمة قفزة بعيدة تستبدل محتويات كل من قطعة الشفرة CS و التسجيل IP بـ 4 بايت تتبع شفرة التشغيل.

شفرة التشغيل	IP	IP	CS	CS
EA	بايت منخفض	بايت عالي	بايت منخفض	بايت عالي

قفزة بعيدة إلى قطعة مختلفة. يتأثر كلا التسجيلان IP و CS . إزاحة هدف -> IP ؛ قطعة هدف -> CS

نظرا لأن قطاع الاقلاع يستخدم كبرنامج عند بدء تشغيل الحاسوب، أول 3 بايت في هذا القطاع ستكون تعليمة القفزة (اللامشروطة) التي، كما ذكرنا سابقا، وظيفتها تجاوز منطقة البيانات التي لا تقبل التنفيذ (أي تجاوز EBPB/BPB) إلى موقع آخر في البرنامج وتنفيذ التعليمات الموجودة هناك (حيث توجد شفرة الاقلاع، عادة في نفس القطاع 0، في المنطقة المحجوزة داخل وحدة التخزين)، في أنظمة إنتل x86، يتم تنفيذ هذه التعليمة، بعد أن يتحول تنفيذ المعالج من MBR إلى قطاع إقلاع القسم. هذه التعليمة بلغة الآلة عادة تأخذ إحدى الصيغتان في أنظمة مايكروسوفت:

<table>
<tr><th colspan="8">تعليمة القفزة (اللامشروطة)</th></tr>
<tr><td colspan="5">منذ دوس 2.0، الأقراص التي تقبل الاقلاع في أنظمة x86 يجب أن تبدأ إما بقفزة قصيرة متبوعة بتعليمة لا عملية NOP
كما تظهر منذ دوس 3.0 (و على أيضا دوس 1.1) في متتالية شفرة التشغيل التالية :</td><td colspan="3">قفزة قصيرة
قفزة قصيرة + بايت إزاحة + تعليمة لا عملية</td></tr>
<tr><td colspan="5"></td><td>0x90</td><td>0x??</td><td>0xEB</td></tr>
<tr><td colspan="5">أو قفزة قريبة كما تبدو في شفرة التشغيل، على معظم أقراص تلفيديو TeleVideo وكومباك Compaq، المهيئة في دوس 2 x وكذلك على بعض أقراص دوس 3.1 إبسون Epson، وأوليفتي Olivetti :</td><td colspan="3">قفزة قريبة (قفزة مباشرة)
قفزة قريبة + 2 بايت إزاحة نسبية</td></tr>
<tr><td colspan="5"></td><td>0x??</td><td>0x??</td><td>0xE9</td></tr>
<tr><td colspan="5">أمثلة في دوس/ ويندوز</td><td colspan="2">وظيفة</td><td>بايت</td></tr>
<tr><td>NTFS
(JMP 0052 NOP)</td><td>EB</td><td rowspan="3">FAT32
(JMP 0058 NOP)</td><td>EB</td><td rowspan="3">FAT12/FAT16
(JMP 003C NOP)</td><td>EB</td><td>تعليمة القفزة</td><td rowspan="2">قفزة قصيرة</td><td>1</td></tr>
<tr><td>52</td><td>58</td><td>3C</td><td>إزاحة نسبية</td><td>1</td></tr>
<tr><td>90</td><td>90</td><td>90</td><td colspan="2">تعليمة لا عملية NOP.</td><td>1</td></tr>
</table>

- الثلاثة بايت في تعليمة القفزة تفكك إلى JMP SHORT 0x?? NOP (لحظ قيمة البايت 0x?? ستكون مختلفة، وتعني أن أي قيمة من 8-بت يمكن أن تكون في هذا البايت)
- الخيارات الصالحة في البايت الأول هي إما EBh (قفزة قصيرة)، أو E9h (قفزة قريبة).

هوية صانعي القطع الأصلية (أو النظام المستخدم في التهيئة)

هذه الجدول يعرض جزء فقط من لائحة هوية صانعي القطع الأصلية OEM ID مع إصدارات دوس المقابلة والمستخدمة في تهيئة قطاع الاقلاع، علما أن أكثر من مصدر يمكن أن ينتج نفس الهوية.

نظام التهيئة	هوية صانعي القطع الأصلية	
كومباك دوس 3.31	IBM 3.3	
أنظمة IBM PC-DOS.	IBM n.m	
نظام تشغيل OS/2	IBM 20.0	
مايكروسوفت OS/2.	OS2 n0.m	
مايكروسوفت-دوس قبل الإصدار 4 وبندوز 2000 (على أقراص FAT16 و FAT32)	MSDOS5.0	
وبندوز 95.	MSWIN4.0	
وبندوز 95 OSR2 أو وبندوز 98	MSWIN4.1	
مايكروسوفت دوس و وبندوز أن تي.	MSWINn.m	
وبندوز 2000 (على أقراص NTFS) / أداة إدارة القرص في وبندوز أن تي 2003	NTFS	
نظام نوفيل دوس Novell DOS.	NWDOSn.m	
نظام دي آر-دوس DR-DOS.	DRDOSn.m	
في لينكس باستخدام mformat من أدوات mtools.	MTOOLn.m	
في لينكس باستخدام mkdosfs من حزمة dosfstools.	Mkdosfs	
وسيلة التهيئة في PC Tools	PC Tools	
التهيئة في TAU.	TAU n0.m	TAU n.m
التهيئة في RxDOS.	RxDOSn.m	

حسب رقم الإصدارة = (n.m)

كتلة معاملات القرص EBPB/BPB

كما ذكرنا سابقا، سجل إقلاع القسم (أو بالتحديد قطاع الاقلاع 0) في المنطقة المحجوزة، يتضمن شفرة (ابتدائية) للبرنامج الاقلاع وبيانات. تأتي مختلطة، البيانات التي ليست شفرة تتضمن معلومات عن المعاملات الفيزيائية الخاصة بوحدة التخزين الحالية، وتعرف باسم كتلة معاملات البيوس BIOS Parameter Block رغم أنها ليست لها علاقة [11] بنظام BIOS

الوثائق الخاصة بمعايير: ECMA-107 و ISO/IEC 9293 (التي تمثل FAT في أقراص التخزين المرنة FDC والوضوئية ODC) تصف هذه الكتلة أيضا لكن باسم: واصف (خرطوشه) [12] القرص المرن FDC descriptor أو واصف (خرطوشه) القرص المرن الممتد FDC extended descriptor. لمعلومات أكثر راجع النشرة السابقة (التي بحجم 259 285 بايت) وهذه المقالة FAT BPB.

كتلة BPB لم تستخدم في الإصدارة الأولى من مايكروسوفت دوس 1. تلك الإصدارة كانت تستخدم فقط بنيتان، إحداهما للأقراص المرنة بوجه واحد والأخرى للأقراص المرنة بوجهين (سعة 360 كيلوبايت، 5.25 بوصة). تحديد النوع على القرص كان يتم بواسطة تفحص أول بايت من بنية FAT1 (تحديدا، 8 بت السفلى) لكن في مايكروسوفت دوس 2، توقف دعم تلك الطريقة القديمة، وحل محلها استخدام معاملات BPB في قطاع الاقلاع، حيث أصبحت جميع وحدات تخزين FAT تتضمن في قطاع الاقلاع BPB. لكن ماهي بنية هذه المعاملات؟، الجواب في الجداول التالية (أنظر أدناه).

في قطاع إقلاع مايكروسوفت دوس 2، تستخدم كتلة BPB فقط من أجل وحدة التخزين FAT التي تملك أقل من 65,536 قطاع (أي 32 ميغابيت مع 512 بايت لكل قطاع). هذا التقييد سببه حجم حقل عدد القطاعات الاجمالية الذي كان فقط 16 بت. لكن نظام مايكروسوفت دوس 3، عالج ذلك القيد عن طريق تضمين حقل جديد في كتلة BPB يدعى حقل عدد القطاعات الإجمالية 32 بت.

كتلة BPB تم تعديلها مرة أخرى في نظام وبندوز 95 مع نظام الملفات الجديد أنذاك FAT32، الذي عالج مشكلة مساحة وحدة التخزين FAT على القرص. لأن نظام ملفات FAT16 كان مقيد بحجم FAT وحجم العنقود في وحدة التخزين التي لا يمكنها أن تتعد 2 جيجابايت باستخدام 512 بايت في حجم القطاع على القرص. علما أن FAT32 BPB تتفق مع FAT12/FAT16 BPB في جميع الحقول حتى حقل عدد القطاعات الاجمالي 32 بت. لكنها تختلف وفقا لنوع الوسيط المستخدم أي FAT12/FAT16 أو FAT32. مع بداية الحيد 24h. (كما سوف تلحظ في الجداول التالية).

للحصول على أقصى توافق لوحدة تخزين FAT وللتأكد أن مشغلات نظام الملفات سوف تفهم وتدعم وحدة التخزين بالشكل الصحيح، ينبغي لكتلة BPB في قطاع إقلاع وحدة التخزين FAT أن تتضمن دائما جميع حقول المعاملات الجديدة سواء كان نوعها FAT12/16 BPB أو FAT32 BPB. كما سوف تلحظ أيضا في شروح التالية.

بالنسبة لكتلة معاملات القرص DPB وتسمى أحيانا كتلة معاملات الوسيط MPB تشبه كتلة معاملات البيوس BPB باستثناء أن التطبيقات في الذاكرة ينبغي أن تصل إلى جدول كتلة DPB للحصول على معلومات القرص المنطقي (وحدة التخزين) بدلا من قطاع الاقلاع.

كتلة DPB عبارة عن جدول بيانات وظيفته تحديد موقع وتخطيط بنى البيانات الأساسية على القرص من أجل نظام الملفات، الجدول يتضمن معلومات محددة عن القسم تستخدم من قبل نظام التشغيل، مثل مواصفاته (كحجم، وعدد القطاعات التي يحتويها..الخ)، ولصقة (اسم)، وعدد القطاعات لكل عنقود على القسم، والبنى الداخلية الأخرى للقسم، مثل جداول توزيع الملفات FATs.

أنظمة التشغيل دوس و OS/2 تحتاج إلى بيانات جدول DPB للتحقق من سعة وحدة التخزين على القرص وموقع البنى المهمة، مثل جدول توزيع الملفات FAT على وحدات تخزين FAT أو جدول الملف الرئيسي MFT على وحدات تخزين NTFS.

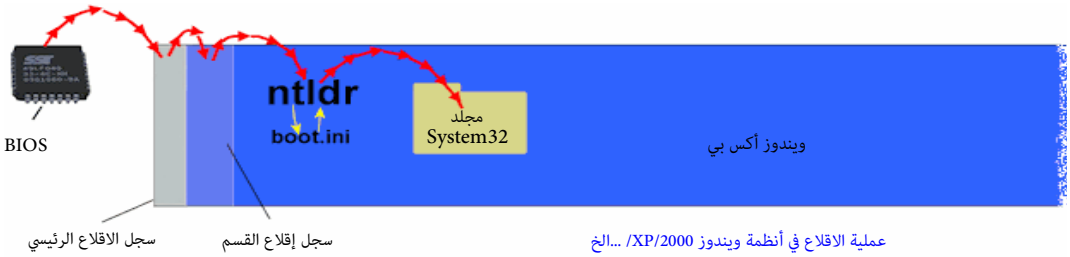
شكل هذه البيانات سيكون خاص جدا (أي مختلف في كل نظام). بالرغم من أن جميع سجلات VBRs تتضمن شفرة إقلاع إلى جانب BPB وبنى أخرى، يتم فقط تنفيذ شفرة إقلاع سجل إقلاع القسم في وحدة التخزين القابلة للإقلاع (أي التي تحمل علم الاقلاع). أما البيانات الأخرى فتقرأ من قبل نظام التشغيل أثناء بدأ التشغيل (وبندوز) لتحديد معاملات وحدة التخزين.

معلومات جدول DPB

- معلومات DPB التي تصف القرص و وحدة التخزين في القرص، يمكن الحصول عليها عن طريق نداءات وظيفة دوس 32H و 1FH. هذه النداءات تعود بمعلومات تستفيد منها الخدمات والتطبيقات التي تنفذ إلى القرص الثابت على مستوى القطاع والمدعوم من مشغلات العتاد.
- بعض الأقراص خصوصا التي لا تقبل الاقلاع وتعمل فقط عن طريق واجهة مشغل العتاد بها. قد لا تتضمن على المعلومات الصحيحة في قطاع الاقلاع وجدول الأقسام، هذا يصعب من عملية تحديد موقع مثلا: حجم الدليل الجذر أو عدد نسخ FATs، ...الخ. لهذا سجل DPB يتضمن جميع هذه المعلومات في شكل بنية واحدة. عمليا جميع المعلومات يمكن الحصول عليها عن طريق قراءة قطاع الاقلاع وتنفيذ نداءات دوس الأخرى مع بعض الحسابات، لكن جدول DPB لديه كل هذه البيانات في مكان واحد.
- 32H في دوس ربما هي الطريقة الوحيدة لإيجاد عنوان مشغل العتاد. أما 1FH فتعود بالمؤشر الخاص بالقرص المبدئي الحالي. لكن هذه النداءات تغير أيضا قيمة التسجيل DS.
- وظيفة دوس 53H. الغير موثقة (مدونة)، تستخدم في دوس أثناء بدء التشغيل لتحضير DPB. هذه الوظيفة 53H. تستطيع ترجمة كتلة BPB (التي يوفرها مشغل العتاد) إلى كتلة DPB. هذه الوظيفة ليس لها أية استخدم عملي مع التطبيقات، لكن قد تستخدمها أداوت القرص أو مشغل جهاز الكتلة الذي ينصب نفسه عن طريق سطر أوامر دوس.

شفرة إقلاع القسم

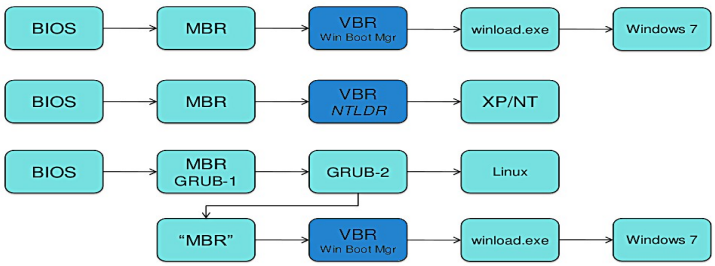
هذه تعليمات تستخدم لتحديد موقع نواة نظام التشغيل الابتدائية وتحميلها أو تحميل ملف برنامج البدء من الدليل الجذر، (في نظام دوس/ ويندوز ME/9x عادة يكون الملف IO.SYS وفي ويندوز أن تي/2000/ XP يكون الملف NTLDR، و في ويندوز فيستا/7 ستكون ملفات winload.exe، BCD، bootmgr).



هذه الشفرة المضمنة في VBR تخص نظام التشغيل الذي يستخدم القسم، وتستخدم لبدأ تحميل نظام التشغيل. ويتم استدعاءها عند إقلاع القرص، إما مباشرة من قبل واجهة البرنامج الثابت للجهاز BIOS (المخزن في رقاقة ذاكرة للقراءة فقط على اللوحة الأم) أو تنفيذها بشكل غير مباشر من قبل شفرة الاقلاع الرئيسية المضمنة في سجل MBR (أو مدير الاقلاع).

في ويندوز تستخدم هذه الشفرة لتحميل القسم الأولي النشيط (أي قسم النظام)، والأقسام الأخرى لا تستخدمها. لكن، كلتا الشفرتان في MBR و VBR يتم تحميلها بنفس الطريقة. للأسف هذا يجعل قطاع الاقلاع مستهدف أيضا من كتاب الفيروسات (راجع: كتيب MBR).

عملية استدعاء أو تنفيذ سجل إقلاع القسم عن طريق محمل أو مدير الاقلاع تعرف باسم chain-loading (أي التحميل بربط الشفرة). بعض أنظمة التشغيل المزدوجة (على الأجهزة المتعدد الأنظمة) مثل محمل الاقلاع NTLDR المستخدم في جميع أنظمة مايكروسوفت المشتقة من ويندوز أن تي، وتشمل ويندوز أكس بي وخادوم ويندوز 2003، تأخذ نسخ من شفرة الاقلاع التي تنصبها أنظمة التشغيل داخل سجل إقلاع القسم وتخزنها في ملفات على القرص، من أجل تحميل محتوى VBR المعني من الملف أثناء الاقلاع (عندما يعرض محمل الاقلاع قائمة أنظمة التشغيل على المستخدم). علما أن في ويندوز فيستا، وخادوم ويندوز 2008 والنسخ الأحدث، تم استبدال NTLDR بعنصرين جديدان هما برنامج WINLOAD.EXE ومدير إقلاع ويندوز BOOTMGR.



الاقلاع في ويندوز ولينكس (أنظمة التشغيل المزدوجة)

برامج وملفات محملات الاقلاع المختلفة.... :	GNU GRUB 2, BOOTMGR, winload.exe, NTLDR
مرحلة الإقلاع الأولي (شفرة محمل الاقلاع في MBR) :	GRUB-1 = GRUB stage 1
مرحلة الإقلاع الثانية (شفرة محمل الاقلاع غالبا توجد في نظام الملفات/نظام التشغيل) :	GRUB-2 = GRUB stage 2

استدعاء الشفرة (التنفيذ)

شفرة الاقلاع في سجل إقلاع القسم VBR تفترض أو تعمل على أساس أن البرنامج الثابت BIOS قد أتم من جانبه، إعداد بنى البيانات، و المقاطعات، وتهيئة العتاد. الشفرة لا تتوقع وجود أكثر من 32 كيلوبايت من الذاكرة من أجل عمل آلية ما يدعى الفشل الأمن ! fail-safe ؛ إذا احتاجت الشفرة قدر أكبر من الذاكرة ينبغي أن تستعلم عن ذلك من INT 12h، لأن شفرة الاقلاع المسبقة الأخرى (مثل أغطية امتدادات نظام البيوس BIOS extension overlays، أنظمة التشفير، أو محملات الاقلاع RIPL) قد تتواجد كذلك في مكان أخر في الذاكرة (وعادة تخفي نفسها عن قطاع الاقلاع بخفض ذاكرة INT 12h المعلن عنها وفقا ذلك، كي لا يعاد كتابتها من قبل عمليات MBR و VBR).

مواصفة BBS تسمح بحجم 64 كيلوبايت من الذاكرة وتوصي بتعيين عناوين الذاكرة من 0000h:7C00h إلى 0000h:FFFFh كذاكرة داخلية مؤقتة.[1] ولا تتوقع شفرة الاقلاع وجود معالجات أفضل من Intel 8088 أو 8086 (المستخدمة في أجهزة الحاسوب الشخصي الأصلية) أو تفترض شيء آخر مهما كانت حالة العتاد، أو نظام المقاطعات (الممكن تمكينها أو تعطيلها) أو موقع وحجم الرصة.

رغم أن النظام IBM BIOS يبدأ تسجلات SS، DS، ES على القطعة 0000h وبقي الرصة الابتدائية عند SS:SP = 0000h:0400h، هذه حالة لا يعتمد عليها، لأن هذا التقليد لا تتبعه جميع شفرات BIOS و MBR. التسجلات التي لم يأتي ذكرها أدناه، تعتبر غير مستخدمة. والنفاذ المباشر إلى العتاد غير مسموح به عادة. بعد تنصيب جدول معاملات القرص DPT/FDPB في الذاكرة عند 0000h:0078h، سجل VBR يجب أن يحرك (وربما يضبط) جدول DPT الذي يشير إليه متجه المقاطعة [13] INT 1Eh في هذا الموقع (INT 1Eh هو مؤشر بعيد إلى DPT وليس مقاطعة).

أحيانا في بعض البيئات الموجهة، بعض محملات الاقلاع الحديثة، تتوقع أن تملك حتى 128 كيلوبايت من الذاكرة من أجل العملية العادية (دون أن تطلب المزيد)، بينما محملات إقلاع أخرى تستخدم LBA في النفاذ للقرص، تتوقع وجود على الأقل معالج 80188 أو 80186.

يتم تحميل سجل VBR عند موقع الذاكرة 0000h:7C00h مع تنصيب تسجيلات المعالج التالية عندما يقوم محمل الإقلاع المسبق (عادة BIOS أو MBR، لكن الاحتمال أن يكون محمل إقلاع آخر) بتمرير عملية التنفيذ إليه عن طريق القفز إلى العنوان 0000h:7C00h في النمط الحقيقي للمعالج.

- CS:IP = 0000h:7C00h (ثابت)

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

- DL = وحدة قرص الإقلاع (رقم جهاز الإقلاع)
- الأقراص المثبتة / الأقراص القابلة للإزالة : الأول = 80h ، الثاني = 81h حتى .... Feh
- الأقراص المرنة /أقراص superfloppies : الأول = 00h، الثاني = 01h حتى ... 7Eh. القيم 7Fh و FFh محجوزة من أجل الأقراص عن بعد/ ROM، ولا يجب استخدمها على القرص.

مثل معظم أنظمة BIOS الأخرى، أنظمة IBM BIOS تدعم أيضا التسجيل DL. لكن نظام توشيبا Toshiba T1000 BIOS لا يدعم DL بالشكل الصحيح، بعض أنظمة Wyse 286 BIOS القديمة تستخدم قيم أكبر أو تساوي 2 في DL من أجل الأقراص الثابتة.

عادة، أقراص الذاكرة USB sticks التي تم إعدادها كأقراص superfloppies تحصل على قيم الإسناد DL = 00h أو DL = 01h. لكن، بعض أنظمة BIOS النادرة تعرض بالخطأ أقراص USB sticks التي تم إعدادها كأقراص قابلة للإزالة تحت التسجيل DL = 01h، بدلا من استخدام DL = 80h.

تقليديا فقط القيم 80h و 00h يتم تمريرها من قبل BIOS أثناء الإقلاع كقيم لأقراص فيزيائية. على كل حال، الكثير من قطاعات الاقلاع مبرمجة ضمنا hard-wired لتعمل مع قيم ثابتة.

مواصفة PnP BIOS و BBS تسمح أيضا بإقلاع الأجهزة الأخرى [2] [1] المواصفة الأخيرة توصي أيضا باستخدام DL من قبل شفرات MBR و VBR بدلا من استعمال القيمة الاعتيادية المضمنة[1]

> شفرة VBR في أنظمة MS-DOS/PC DOS ,OS/2 تتجاهل قيمة DL المقدمة وتبحث عن القيمة المخزنة عند الحيد 19h+ في كتلة EBPB أو عند حيد القطاع 1FDh+ في نسخ دوس 3.2 حتى النسخة 3.31.
>
> بعض نسخ نظام DR-DOS تستخدم DL منذ الإصدار 7.02. سجلات VBR في نظام DR-DOS 7.07 تستخدم DL وتتجاهل قيمة BPB، لكن يمكنها استخدام الخيار [nnn:] SYS /O لإعادة استدعاء الطريقة القديمة في استخدام قيمة BPB أو حتى فرض قرص إقلاع معين nnn لتخزن هناك. نظام FreeDOS يستفيد أيضا من قيمة DL في وحدات التخزين FAT32، بينما يفعل ذلك في وحدات تخزين FAT12/FAT16 فقط.إذا تم تعيين قيمة BPB إلى FFh.

- DH : بت 5 = 0

الجهاز المدعوم في BIOS من خلال INT 13h ؛ ما عدا ذلك: لا تهم (ينبغي أن تكون صفر) بعض أنظمة IBM BIOS تدعم DH. وبعض شفرات MBR و VBR تحفظ قيمة DH.

الأنظمة التي تدعم تقنية القيس والتشغيل PnP BIOS أو مواصفة BBS سوف توفر مؤشر إلى بيانات PnP بالإضافة إلى DL : [2][1]

- 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 غطاء PnP / BBS overlay المقيم في الذاكرة من أجل تضبيط ترتيب الإقلاع..الخ، لكن هذه المعلومات يتم تجاهلها من قبل معظم السجلات المعيارية MBR و VBR.

تسجيلات ES:DI يتم تمريرها بشكل جيد إلى VBR، لكن أنظمة التشغيل التي تستخدم تقنية PnP عادة تملك أيضا طرق احتياطية للاسترداد مدخلة PnP BIOS في ما بعد، لذلك معظم أنظمة التشغيل لا تعتمد على هذا. المعلومات في ES:DI يمكن استخدامها كإشارة وفقا لمواصفة PnP BIOS، يمكن إيجاد بنية تفحص تنصيب "$PnP" بالبحث عن توقيع سلسلة أسكي "$PnP" في ذاكرة النظام بداية من F0000h وحتى FFFFFh عند كل حد 16 بايت..

في الوسيط المقسم، عند تنفيذ VBR من قبل MBR (أو محمل إقلاع آخر) بدلا من BIOS، عدة تطبيقات تمرر معلومات إضافية إلى VBR إلى جانب فقط DL (وأحيانا أيضا DH و ES:DI):

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

نظام التشغيل PC-MOS 5.1 يعتمد على هذا في الإقلاع إذا لم يتم تعيين قسم للإقلاع (قسم بـ علم إقلاع) في جدول الأقسام.

قطاعات إقلاع أنظمة Multiuser DOS و REAL/32 تستخدم هذا مع وسيلة الإقلاع LOADER، في تحديد موقع قطاع إقلاع القسم النشيط (أو محمل إقلاع أخر مثل IBMBIO.LDR في موضع ثابت على القرص) إذا لم تعثر على ملف الاقلاع LOADER.SYS.

أنظمة PTS-DOS 6.6 و S/DOS 1.0 تستخدم هذا مع ميزة (وظيفة) القسم النشيط المتقدم AAP. بالإضافة إلى دعم وسيلة الإقلاع LOADER وأقسام AAP، أنظمة DR-DOS 7.07 يمكنها استخدام هذا في تقرير أسلوب نفاذ INT 13h الضروري عند استخدام شفرتها المزدوجة CHS/LBA VBR.

شفرة MBR في أنظمة OS/2 و MS-DOS (قبل النسخة 7.0) و PC DOS 2.0 (حتى 7.10) و ويندوز أن تي (حتى 2007)، توفر أيضا نفس هذه الواجهة رغم أن تلك الأنظمة لا تستخدمها.

شفرة MBR في أنظمة ويندوز أن تي 6.0 (والنسخ الأحدث) تستخدم تسجيلات المعالج الأخرى، ولذلك لم تعد متوافقة مع هذه الامتدادات (لم تعد توفر المؤشر DS:SI) في حين أن بعض الامتدادات تعتمد فقط على مدخلة 16-بايت في جدول الأقسام نفسها، نجد امتدادات أخرى قد تتطلب تمثيل (حضور) كافة مدخلات جدول الأقسام 4 (أو 5) كذلك.

في نظام DR-DOS 7.07، سجل MBR مع LOADER يمكنه أن يستخدم اختياريا الواجهة الممتدة :

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

عند استعمال مخطط تقسيم القرص GPT، اللجنة الفنية T13 المسؤولة عن معايير واجهة ATA تقترح شفرة سجل إقلاع رئيسي هجين Hybrid MBR مع المواصفة الرابعة لمحرك الأقراص المحسن EDD-4 (مواصفة قرص ظاهري) هذا الاقتراح يوصي بامتداد آخر إلى الواجهة بين سجل الاقلاع الرئيسي وسجل إقلاع القسم [3] VBR ↔ MBR:

- EAX = 54504721h بمعنى "GPT !"

ويشير إلى أن بنية تسليم سجل الاقلاع الرئيسي الهجين hybrid MBR قد تم تمريرها مع DS:SI عوضا عن سجل القسم التقليدي في MBR.

- DL = وحدة قرص الاقلاع (رقم جهاز الاقلاع) (انظر أعلاه)
- ES:DI = تشير إلى بنية تفحص تنصيب "$PnP" (انظر أعلاه)
- DS: SI = تشير إلى بنية تسليم سجل الاقلاع الرئيسي الهجين hybrid MBR، التي تتألف من المدخلة الافتراضية 16-بايت في جدول أقسام MBR.

(ستكون جميع البتات في حالة تعيين باستثناء علم الاقلاع عند الحيد 0h+ ونوع القسم عند الحيد 4h+)، متبوعة ببيانات إضافية. هذا يتوافق جزئيا مع امتداد المؤشر القديم DS:SI المذكور أعلاه. إذا كانت فقط مدخلة القسم 16-بايت، وليس كامل جدول الأقسام مطلوب من قبل هذه الامتدادات القديمة.

LOADER.COM يعرف أيضا باسم NEWLDR: هو 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، والبحوث الرقمية/ديجيتال ريسيرش Digital Research...وغيرها...

ملف LOADER.SYS جزء من تنصيب LOADER.COM.

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

توقيع قطاع الاقلاع

يستخدمه نظام BIOS والشفرات الأخرى للتحقق من صحة قطاع الاقلاع.

في الأجهزة المتوافقة مع أنظمة IBM PC حضور محمل الاقلاع في قطاع إقلاع x86، يعبر عنه رسميا بالشفرة الست العشرية 16-بت AA55h وتدعى توقيع قطاع الاقلاع (55h عند الحيد 1FEh و AAh عند الحيد 1FFh) في قطاع الاقلاع الذي بحجم 512 بايت أو أكثر [1]. هذا التوقيع أيضا يحدد نهاية القطاع في قطاعات 512 بايت. سجلات VBRs التي على القطاعات الأصغر أو الأكبر يمكن أن تعرض كذلك التواقيع عند نهاية حجم القطاع الفعلي، لكن المعلومات الواردة هنا تطبق فقط على توقيع 16-بت عند الحيد 1FEh.

هذا التوقيع يشير إلى وجود على الأقل محمل إقلاع واحد افتراضي يمكن تنفيذه بأمان، حتى وإن كان في الواقع غير قادر على تحميل نظام التشغيل. هذه الشفرة إذا لم يعثر عليها نظام BIOS أو MBR يعرض رسالة خطأ ويتوقف تحميل نظام التشغيل. التوقيع لا يشير إلى وجود نظام ملفات أو نظام تشغيل (معين)، رغم أن بعض نسخ دوس القديمة قبل نسخة 3.3 تعتمد عليه في التحري عن الوسيط المهيئ بنظام FAT (لكن النسخ الحديثة لا تفعل ذلك). شفرة إقلاع المنصات أو المعالجات الأخرى لا تستخدم هذا التوقيع، لأن استخدامه يمكن أن يسبب انهيار في النظام عند تمرير نظام BIOS عملية التنفيذ إلى قطاع الاقلاع الذي يفترض أنه يتضمن شفرة تنفيذية صالحة. هذا يفترض من وسائط FAT12/FAT16 المستخدمة أيضا من قبل نسخ دوس القديمة جدا أن تحفظ التوقيع حتى وإن كانت لا تتضمن نظام تشغيل أو قصد منها أن تقبل الاقلاع على المنصات الأخرى فقط؛ ولذلك يجب أن تتضمن كذلك على الأقل محمل إقلاع واحد (افتراضي) متوافق مع أنظمة x86 (للمقارنة راجع أمثلة FAT في وسائط Atari ST و MSX-DOS). رغم ذلك، بعض وسائط المنصات الأخرى تتضمن بالخطأ التوقيع بدون حتى وجود محمل افتراضي متوافق مع أنظمة x86 ، مما يجعل عملية التحقق غير موثوقة 100%.

باستثناء أجهزة الحاسوب الشخصي الأصلية IBM PC وبعض الأجهزة الأخرى، معظم أنظمة BIOS تتفحص هذا التوقيع، منذ (على الأقل) IBM PC/AT. أيضا، معظم شفرات محملات الاقلاع في سجل MBR تتفحص هذا التوقيع قبل تمرير التحكم إلى قطاع الاقلاع. بعض أنظمة BIOS (مثل IBM PC/AT) تتفحص فقط الأقراص المثبتة / الأقراص القابلة للإزالة، بينما الأقراص المرنة وأقراص superfloppies يكفي أن تبدأ ببايت أكبر أو يساوي 06h وأن لا تتضمن كلمات (2-بايت) التسعة الأولى نفس القيمة، قبل القبول بصحة قطاع الاقلاع، حتى يتم تجنب فحص AAh، 55h على الأقراص المرنة. وبما أن قطاعات الاقلاع القديمة (مثل وسائط دوس و CP/M-86) أحيانا لا تملك هذا التوقيع رغم أنها تستطيع الاقلاع، يمكن تعطيل عملية التفحص في بعض البيئات. هذا يعكس أيضا حقيقة إمكانية تهيئة الأقراص المرنة كي تستخدم قطاعات بأحجام أصغر من 512 بايت.

إذا شفرة نظام BIOS أو MBR لم تكتشف قطاع إقلاع صالح وبالتالي لم تستطيع تمرير عملية التنفيذ إلى شفرة قطاع الاقلاع، سوف تحاول مع جهاز الاقلاع التالي في ترتيب الأجهزة الموجودة. إذا فشلت جميعها تعرض رسالة خطأ وتستدعي INT 18h. هذا سوف إما يبدأ في تشغيل البرمجية المقيمة اختياريا في ذاكرة ROM BASIC، أو محاولة الاقلاع عن بعد عبر الشبكة، أو إعادة تشغيل النظام عن طريق INT 19h بعد التأكيد من المستخدم، أو يجعل النظام يوقف عملية الاقلاع حتى يتم استخدام مفتاح تشغيل الحاسوب الخارجي مرة أخرى.

نظرة على أنظمة الملفات

في الأنظمة التالية سيكون التركيز فقط على قطاع الاقلاع. (شرح أنظمة الملفات بدون الخوض في التفاصيل).

	نظام الملفات	نظام التشغيل		معالج/ منصة	حاسوب منزلي	مصدر البرمجية	مصدر العتاد
1	(FAT12, FAT16, FAT32)	مايكروسوفت دوس/ويندوز		X86 (Intel)	حاسوب شخصي	مايكروسوفت	متعددة
2	NTFS	مايكروسوفت ويندوز					
3	FAT12 (فقط) أو FAT16 (رقع patches)	MSX-BASIC / MSX-DOS		Zilog Z80	MSX	مايكروسوفت	ASCII
4	FAT12 معدل / FAT16	Atari TOS		Motorola 680x0	Atari ST	Atari	Atari

نظام ملفات FAT

جدول توزيع الملفات أو جدول تخصيص الملفات FAT، نظام ملفات غير معقد، صمم في الأصل للاستخدام على الأقراص الصغيرة من أجل بنى الأدلة البسيطة.

التسمية تعود إلى طريقة تنظيم وتخزين نظام الملفات للمدخلات في جدول يشبه فهرس الكتاب يقع في بداية وحدة التخزين. يستخدمه نظام التشغيل عند البحث عن ملف معين ومعرفة في أية عناقيد على القرص مكتوب ذلك الملف.

في أنظمة FAT12/16، جداول توزيع الملفات والمجلد الجذر يجب تخزينها في موقع ثابت حتى يستطيع النظام تحديد موقع الملفات المطلوبة عند بدء التشغيل، وفي وحدة تخزين FAT32 الدليل الجذر موجود في منطقة البيانات، أي ليس في موقع ثابت وليس بحجم ثابت.

وحدة التخزين المهيئة بنظام ملفات FAT تكون موزعة على عناقيد. الحجم المبدئي للعنقود يحدد بناء على حجم وحدة التخزين.

نظام ملفات FAT12

(يستخدم في معظم الأقراص المرنة، 3.5 في أنظمة x86،)

نظام ملفات FAT12 مصمم أصلا لأقراص المرنة، ولا يستطيع التعامل مع حجم أكبر من 16 ميغابايت لأنه يستخدم 12 بت في معالجة العناقيد (وحدات التخزين الأصغر من 16 ميغابايت يجب أن تأخذ تهيئة FAT12).

نظام ملفات FAT16

(يستخدم في أنظمة مثل مايكروسوفت دوس، ويندوز 95/98 - الأقراص الصغرى)

نظام ملفات FAT16 صمم للأقراص الثابتة القديمة، ولا يستطيع التعامل مع حجم عناقيد أكبر من 64 كيلوبايت. ونظرا لأن في الأقراص الثابتة الكبرى حجم العنقود سيكون أكبر، هذا سوف ينتج عنه مساحة كبيرة مهملة على القرص (تدعى: slack space).

القطاع الأول (قطاع الاقلاع) يحتوي على معلومات تستخدم في حساب أحجام ومواقع المناطق الأخرى. قطاع الاقلاع يتضمن أيضا شفرة لإقلاع نظام التشغيل المنصب في وحدة التخزين.

منطقة البيانات موزعة على كتل منطقية تدعى عناقيد (أو وحدات تخصيص). كل عنقود منها يملك مدخلة مصاحبة في منطقة جدول توزيع الملفات FAT.

المدخلة المخصصة للعنقود تتضمن إما قيمة العنقود التالي الذي يتضمن بيانات من الملف، أو تتضمن ما يسمى قيمة نهاية الملف EOF والتي تعني لا وجود لعناقيد إضافية تتضمن بيانات من الملف. الدليل الجذر والأدلة الثانوية تتضمن اسم الملف والتواريخ، وأعلام الخاصة، ومعلومات عنقود البداية التي تخص كائنات نظام الملفات.

بنية نظام الملفات الأساسية في FAT16
منطقة القطاعات المحجوزة، (عند البداية وتتضمن قطاع إقلاع واحد فقط في FAT12/16)
منطقة جدول توزيع الملفات FAT
منطقة الدليل الجذر
منطقة البيانات

اسم الحقل (في قطاع إقلاع FAT16)	حجم (بايت)	إزاحة
تعليمة القفزة	3	000h (0)
هوية صانعي القطع الأصلية OEM ID	8	003h (3)
معاملات كتلة BPB	25	00Bh (11)
معاملات الكتلة الممتدة EBPB	26	024h (36)
شفرة إقلاع ابتدائية	448	03Eh (62)
توقيع القطاع (علامة نهاية القطاع)	2	1FEh (510)

من محاسن FAT16 :

- نظام الملفات FAT16 يستخدم في أنظمة ويندوز 98/95/NT/2000 وفي بعض أنظمة يونكس.
- هناك العديد من البرمجيات المستخدمة في معالجة المشاكل واستعادة البيانات على وحدات تخزين FAT16.
- إذا واجهتك مشكلة في بدء التشغيل، يمكنك تشغيل الحاسوب باستخدام قرص مرن لنظام مايكروسوفت دوس.
- نظام الملفات FAT16 فعال ومناسب للاستخدام على وحدات التخزين الأصغر من 256 ميغابايت (من حيث التخزين والسرعة).

من مساوئ FAT16 :

- المجلد الجذر يمكن أن يتعامل فقط مع 512 مدخلة كحد أقصى. استخدام أسماء الملفات الطويلة (LFN) يمكن أن يخفض بشكل ملحوظ عدد المدخلات المتوفرة.
- نظام ملفات FAT16 مقيد بـ 65.536 عنقود، لكن بسبب بعض العناقيد المحجوزة، الحد العملي للعناقيد هو 65.524 عنقود. وحدة التخزين FAT16 الكبيرة في ويندوز 2000 مقيدة بـ 4 جيجابايت وتستخدم حجم عنقود 64 كيلوبايت. وحتى تبقى متوافقة مع أنظمة مايكروسوفت دوس، و ويندوز 95/98، لا يمكن لوحدة التخزين أن تكون أكبر من 2 جيجابايت.
- نظام ملفات FAT16 ليس فعال على وحدات التخزين بحجم كبير، لأن حجم العنقود يمكن أن يزيد. المساحة المخصصة لتخزين الملف ترتكز على حجم العنقود المخصص، وليس حجم الملف، مثلا في حالة : ملف 10 كيلوبايت مخزن على وحدة تخزين 1.2 جيجابايت، تستخدم عنقود 32-كيلوبايت، ستكون المساحة الضائعة من القرص 22 كيلوبايت.
- لا توجد نسخة احتياطية من قطاع الاقلاع.
- لا يوجد نظام للتأمين نظام الملفات مدمج أو آلية للضغط البيانات في FAT16.
- وحدات التخزين FAT16 الأكبر من 2 جيجابايت لا يمكن النفاذ إليها من أجهزة حاسوب تستخدم مايكروسوفت دوس، وويندوز 95/98، وعدة أنظمة أخرى.

نظام ملفات FAT32

نظام ملفات من زمن و ويندوز 95/98، يستطيع التعامل مع حجم عناقيد أكبر من 64 كيلوبايت. لكن 4 بت العليا من عنقود 32 بت محجوزة ولا تستخدم أبدا. (لذلك تستطيع تسميته FAT28) وكما يدل اسمه ؛ FAT32 هذا النظام للملفات يستطيع معالجة كحد أقصى 256 ميغابايت للعنقود في كل قسم. الذي يمكن الأقراص الثابتة الكبرى من الإبقاء على أحجام العنقود الصغرى وتقليص المساحة المهملة بين الملفات.:

من محاسن FAT32 :

- المجلد الجذر على قرص FAT32 عبارة عن سلسلة عناقيد اعتيادية يمكن أن تتواجد في أي مكان على وحدة التخزين. لهذا السبب، FAT32 غير مقيد بعدد المدخلات في المجلد الجذر.
- يستخدم عناقيد أصغر حجم (4 كيلوبايت لأجل وحدات تخزين تصل إلى 8 جيجابايت)، لذلك، هذا النظام أكثر فاعلية في تخصيص مساحة القرص مقارنة بنظام ملفات FAT16. ووفقا لحجم ملفاتك، عند استخدام FAT32 هناك إمكانية للاستفادة من مساحات إضافية على القرص على وحدات التخزين الكبرى مقارنة بنظام FAT16.
- يستطيع آليا استخدام النسخة الاحتياطية من FAT بدلا من الاعتيادية (مع FAT16، فقط أداوت إصلاح القرص مثل Chkdsk تستطيع توظيف النسخة الاحتياطية).
- يتم آليا عمل نسخة احتياطية من قطاع الاقلاع في الموقع المحدد في وحدة التخزين، لذلك، وحدات تخزين FAT32 أقل عرضة للفشل مقارنة بوحدات تخزين FAT16.

من مساوئ FAT32 :

- أكبر وحدة تخزين FAT32 يستطيع نظام ويندوز تهيئتها هي بحجم 32 جيجابايت.
- لا يمكن النفاذ مباشرة إلى وحدات تخزين FAT32 من أنظمة التشغيل الأخرى باستثناء ويندوز 98/ 95 OSR2 (لكن يمكن النفاذ من لينكس في حالة تعدد الأنظمة ومن ويندوز).
- إذا واجهتك مشكلة في بدء التشغيل، لا يمكنك تشغيل الحاسوب باستخدام قرص مرن نظام مايكروسوفت دوس و ويندوز 95، (باستثناء أصدارة OSR2 أو اللاحقة).
- لا يوجد نظام للتأمين نظام الملفات مدمج أو آلية للضغط البيانات في FAT32.

الاختلاف بين الإصدارات

نظام ملفات FAT يملك عدة إصدارات مختلفة FAT32، FAT16، FAT12، (بالإضافة إلى النسخ الأخرى المعدلة) وكل إصدارة مصممة لاستخدام في حجم مختلف من وسائط التخزين (الأقراص).

- الأرقام في أسماء FAT12 و FAT16 و FAT32 تشير إلى عدد البتات المطلوبة في مدخلة جدول توزيع الملفات FAT.
- نظام ملفات FAT12 يستخدم مدخلة 12-بت (2^{12} عنقود).
- نظام ملفات FAT16 يستخدم مدخلة 16-بت (2^{16} عنقود).
- نظام ملفات FAT32 يستخدم مدخلة 32-بت، لكن نظام ويندوز يحتفظ بأول 4 بت من مدخلة FAT، هذا يعني أن FAT32 يملك 2^{28} عنقود كحد أقصى.

حدود العنقود	عدد البايت في كل عنقود ضمن FAT	نظام الملفات
عدد عناقيد البيانات أقل من 4087 عنقود	1.5	FAT12
عدد عناقيد البيانات بين 4087 و 65526 عنقود	2	FAT16
عدد عناقيد البيانات بين 65526 و 268,435,456 عنقود	4	FAT32

في وحدة تخزين FAT32، نسخة FAT يمكن أن تكون كبيرة، على عكس نظيرتها في وحدة تخزين FAT16 المقيدة بحجم أقصى 128 كيلوبايت كقيمة للقطاعات (2 بايت * 65526 عنقود = 131052 بايت = 128 كيلوبايت)، ووحدة التخزين FAT12 المقيدة بحجم أقصى 6 كيلوبايت كقيمة للقطاعات (1.5 بايت * 4087 عنقود = 6130.5 بايت = 6 كيلوبايت)، لهذا السبب، يتم تخزين تعداد العنقود الحر "الأخير المعروف" على وحدة تخزين FAT32 حتى لا يضطر إلى حسابه كلما صدرت روتينات نداء API التي تستفسر عن حجم المساحة الحرة على وحدة التخزين.

رقم قطاع معلومات نظام الملفات هو القيمة الموجودة في حقل الكتلة BPB والتي تأخذ دائما القيمة 1 في أنظمة تشغيل مايكروسوفت. (راجع أدناه: جدول قطاع معلومات نظام الملفات FAT32).

بنية وحدة تخزين FAT

نظام ملفات FAT مركب من أربعة أجزاء مختلفة في القسم (وحدة تخزين).

	محتوى		حجم القطاعات (في المناطق الأربعة)	إزاحة
1	قطاع الاقلاع		# القطاعات المحجوزة	بداية القسم
	قطاع معلومات نظام الملفات (فقط في FAT32)			
	قطاعات إضافية محجوزة (اختيارية)			
2	جدول توزيع الملفات # 1 (FAT1)	جداول توزيع الملفات	(# FATs) * (# القطاعات لكل FAT)	بداية القسم + # القطاعات المحجوزة
	جدول توزيع الملفات # 2 (FAT2) (شرطية)			
3	الدليل الجذر [2] (سيكون في مكان وحجم ثابتين في FAT12/FAT16)		(# مدخلات الجذر * 32) / (# بايتات لكل قطاع)	بداية القسم + # القطاعات المحجوزة + (# القطاعات لكل FAT * 2)
4	منطقة البيانات (للملفات والأدلة ... إلى نهاية القسم أو القرص)		(# العناقيد) * (# القطاعات لكل عنقود)	(مع افتراض أن مرآوية [15] FAT في حالة تمكين).

القطاعات المحجوزة

(تقع في بداية وحدة التخزين وتتضمن سجل الاقلاع، الذي يدعى أيضا قطاع الاقلاع، أو كتلة الاقلاع، أو القطاع 0).

أول قطاع في هذه المنطقة سيكون قطاع الاقلاع أو VBR ويتضمن عادة منطقة BPB، مع الجزء الأول أو كامل شفرة محمل إقلاع نظام التشغيل.

في بعض الأنظمة مثل دوس و OS/2، المعلومات الهامة في قطاع الاقلاع يمكن أيضا الوصول إليها من خلال بنية نظام تشغيل تدعى كتلة معاملات القرص DPB. (أنظر للشرح أعلاه)

العدد الإجمالي للقطاعات المحجوزة يشير إليه حقل داخل قطاع الاقلاع. في أنظمة ملفات FAT32 عادة قيمة الحقل تكون 32، تشمل قطاعات النسخة الاحتياطية الثلاثة (أنظر للجداول).

شفرة قطاع الاقلاع في أنظمة مايكروسوفت تخطت القطاعات المنطقية 0 و 1 منذ إصدار FAT32، مع اعتماد القطاع المنطقي 0 على الروتينات الثانوية في القطاع المنطقي 2. رغم ذلك العديد من المنتجين ما زال يستخدم إعدادات القطاع-الواحد من أجل محمل الاقلاع.

إذا تم إنشاء أقسام FAT32 في أنظمة ويندوز أكس بي وويندوز 7، القطاع الثالث في سجل الاقلاع الجديد لن يتضمن أية شفرة؛ وسيكون حشو بايت صفر فقط، باستثناء التوقيع 55h AAh، وسوف يستخدم القطاع المنطقي 12 (أي القطاع 13) بدلا من القطاع الثالث في منطقة القطاعات المحجوزة من أجل محمل إقلاع ممتد (موسع).

نوع قطاع الاقلاع	عدد القطاعات		نسخة احتياطية	أنظمة التشغيل
	المحجوزة	المستخدمة		
FAT12 / FAT16	1	1	لا توجد	دوس، جميع إصدارات مايكروسوفت
FAT32	32	3 من 6	في المنطقة المحجوزة	دوس، مايكروسوفت ويندوز
NTFS	16	6 + 1	في أخر قطاع في القسم	مايكروسوفت ويندوز

منطقة جدول توزيع الملفات FAT

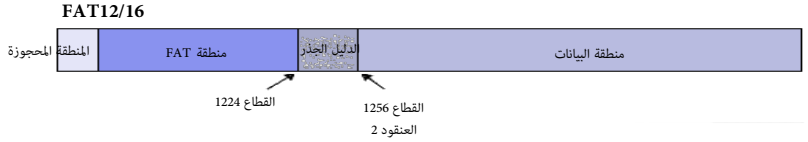
عادة هذه المنطقة تتضمن نسختين من جدول توزيع الملفات (قد تتفاوت) لغرض التدقيق عن الأخطاء، رغم أنها نادرا ما تستخدم، حتى من قبل أدوات إصلاح القرص.

هذه عبارة عن مخططات لمنطقة البيانات، تشير إلى العناقيد المستخدمة من قبل الملفات والأدلة. وتعقب مباشرة القطاعات المحجوزة، في أنظمة ملفات FAT12 و FAT16.

تحفظ النسخ الإضافية بالأخص متزامنة عند الكتابة والقراءة وتستخدم فقط عند حدوث خطأ في أول نسخة من FAT (أي FAT1). في FAT32، يمكن تغيير هذا السلوك واختيار جدول توزيع ملفات واحد من تلك المتوفرة لاستخدامه لأغراض التشخيص. علما أن أول اثنان من العناقيد في مخطط (العنقود 0 والعنقود 1) تتضمن قيم خاصة.

منطقة الدليل الجذر

هذه عبارة عن جدول دليل يخزن معلومات عن الملفات والأدلة التي تقع في الدليل الجذر. المنطقة تستخدم فقط في FAT12 و FAT16، ويفرض على الدليل الجذر حجم أقصى ثابت مخصص مسبقا عند إنشاء وحدة التخزين. لكن FAT32 يخزن الدليل الجذر في منطقة البيانات، إلى جانب الملفات والأدلة الأخرى، هذا يسمح للمجلد الجذر بالتوسع تقريبا بلا قيد. لهذا السبب منطقة البيانات تبدأ هنا في FAT32.



في FAT12/16، العنقود 2 يتبع الدليل الجذر، بينما في FAT32 العنقود 2 هو أول قطاع من منطقة البيانات

منطقة البيانات

منطقة بيانات القسم تأتي بعد سجل إقلاع القسم وتحتل معظم القسم، حيث تخزن الأدلة والملفات الفعلية. هذه المنطقة من القرص مقسمة إلى عناقيد ويديرها نظام الملفات.

المحتوى النوعي هنا يتفاوت لأنه يرتكز على نظام الملفات المستخدم، وترتيب تخزين الملفات، ومستوى إلغاء تجزئة الملفات وهكذا.... في مثل هذه الحالة لا يمكن عرض بنية معينة هنا لأن البنى ستكون ديناميكية، أي تتغير مع تغير الملفات والبيانات على القرص.

حجم الملفات والأدلة الثانوية يمكن أن يزيد (طالما كانت هناك عناقيد حرة) بإضافة روابط أكثر إلى سلسلة الملفات في FAT. لكن بإعتبار أن الملفات موزعة على وحدات من عناقيد، إذا كان هناك ملف بحجم 1 كيلوبايت في عنقود بحجم 32 كيلوبايت، بقية 31 كيلوبايت لا تستخدم، أي أنها مساحة ضائعة. نظام ملفات FAT32 بالأخص يبدأ جدول الدليل الجذر في العنقود رقم 2 : العنقود الأول من منطقة السانات.

تقليديا، الأجزاء الغير مستخدمة من منطقة السانات يتم حشوها بالقيمة 0xF6 وفقا لجدول DPT (مؤشر INT 1Eh)، عند تهيئة الأجهزة المتوافقة مع أنظمة IBM، وكذلك على حاسوب الحبر أتاري بورتفوليو Atari Portfolio. في نظام CP/M الأقراص المرنة 8-بوصات بالأخص تأتي مهيئة مسبقا بالقيمة 0xE5 ؛ ووفقا لشركة Digital Research هذه القيمة كانت تستخدم أيضا على الأقراص المرنة المهيئة في Atari ST. بينما شركة أمستراد Amstrad كانت تستخدم القيمة 0xF4.

بعض الأدوات الحديثة تمسح الأقراص الثابتة بالقيمة 0x00، في حين أن 0xFF، القيمة الاعتيادية للكتلة الذاكرة المدمجة بدون برمجة (non-programmed flash block) وتستخدم على أقراص flash لتقليل من اهتراء القرص wear (دورات المسح P/E) القيمة الأخيرة تستخدم أيضا على أقراص ROM، علما أن بعض أدوات التهيئة المتقدمة تسمح بضبط بايت حشو التهيئة.

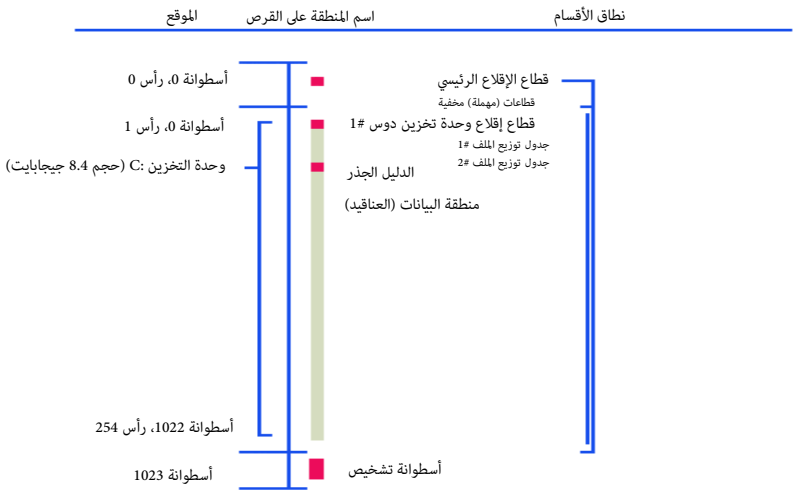
أسطوانة تشخيص (مع خاصية القراءة والكتابة)

في الأنظمة القديمة التي لا تدعم LBA (أي عنونة الكتل المنطقية للنفاذ إلى القرص)، برامج تقسيم القرص مثل FDISK تحتفظ عادة بالأسطوانة الأخيرة على القرص الثابت لاستخدامها كأسطوانة اختبار تشخيصي. بسبب وجود هذا الأسطوانة، FDISK قد يعرض عدد أسطوانات أقل من العدد الفعلي (أي اختلف بين السعة الإجمالية المعلن عنها من قبل FDISK والسعة المعلن عنها من صانع القرص). في هذه الحالة، نظام التشغيل لن يستخدم هذه الأسطوانة لأنها تقع خارج المنطقة المقسمة من القرص.

في الأنظمة التي تستخدم المنطقة المحمية للمضيف HPA، يمكن للنظام الاحتفاظ بحيز في نهاية القرص يستخدم في استرداد النظام أو تستخدمه برمجية استعادة النظام، وأدوات التشخيص الأخرى.

منطقة التشخيص تسمح للبرمجية (مثل قرص التشخيص من المنتج/البائع) القيام باختبارات القراءة والكتابة على القرص الثابت بدون إتلاف بيانات المستخدم. العديد من هذه البرمجيات أيضا يستبدل الأسطوانة المتضررة بأسطوانة إضافية إذا وجدها أثناء الاختبار.

شكل مبسط يظهر العلاقة الفيزيائية التي تربط بين سجل إقلاع القسم وبنى البيانات الأخرى على القرص.



بنية إدارة ملفات FAT16 على قرص قياسي 8.4 جيجابايت. يعرض مواقع MBR و VBR على قرص قسم واحد FAT القرص يدعم ترجمة قياسات القرص CHS باستخدام LBA

بنية قطاعات الاقلاع

- في جداول الأنظمة التالية FAT12 و FAT16 و FAT32 و NTFS، جميع الأسماء التي تبدأ بالمحارف اللاتينية _BPB جزء من معاملات BPB. والأسماء التي تبدأ بـ_BS جزء من قطاع الاقلاع. الكتلتان BPB و EBPB سوف تكون دائما في القطاع 0 في وحدة تخزين FAT و NTFS.
- كتلة BPB تعرف بأسماء عدة : كتلة معاملات نظام الإدخال والإخراج الأساسي (البيوس)، وكتلة معاملات القرص، أو كتلة معاملات الوسيط (كما هو الحال في القرص المرن)، وربما قد تسمى أيضا الوحدة التجميعية لمعاملات البيوس! (رغم أنها ليس لها علاقة إطلاقا بنظام BIOS ولا يستخدمها [11]، والذي يستخدمها هو نظام الملفات (واختياريا خارج دوس).
- قيم الست عشري (البيانات العددية التي تتطلب أكثر من بايت واحد) في أنظمة إنتيل x86 دائما تخزن في الذاكرة بايت الأدنى أولا وبايت الأعلى أخيرا لتسريع عمليات المعالج [24]. بمعنى: الأرقام الممثلة في أكثر من بايت واحد تخزن/تظهر بترتيب نهوي صغير أو بترتيب ثمانيات معكوس، لكن في بعض الأمثلة تظهر نهوي كبير. (هذا بسبب اختلف المصادر) [1].

في بنية قطاع الاقلاع، أول 11 بايت ستكون بنية مشتركة في معظم إصدارات FAT، المتوافقة مع أجهزة IBM وأنظمة x86، منذ دوس 2.0.

وصف	طول/بايت	إزاحة البايت

تعليمة القفزة (اللامشروطة). من أجل القفز إلى موقع آخر في البرنامج (أي تتجاوز معاملات BPB إلى شفرة الاقلاع) وتنفيذ التعليمات الموجودة هناك، يستخدم غالبا في هذا الحقل إحدى الصيغتان:

من دون هذه القفزة، المعالج سوف يحاول تنفيذ البيانات التي ليست شفرة (BPB/EBPB) بعد تحميل القطاع الأول في الذاكرة عند العنوان 0x0000:0x7c00، إذا كان قطاع الاقلاع يملك توقيع صالح في أخر 2 بايت من قطاع الاقلاع (الذي تتفحصه معظم محملات الاقلاع المتواجدة في BIOS أو MBR) وتقلع منه وحدة التخزين، محمل الاقلاع المسبق سوف يمرر التنفيذ إلى هذه المدخلة مع قيم تسجيل معينة، و تعليمة القفزة سوف تتجاوز بقية الترويسة (التي لا تقبل التنفيذ).

- منذ دوس 2.0، الأقراص التي تقبل الاقلاع في أنظمة x86 يجب أن تبدأ إما بقفزة قصيرة متبوعة بتعليمة NOP، (كما تظهر منذ دوس 3.0 و على دوس 1.1 في متتالية شفرة التشغيل أعلاه).
- أو قفزة قريبة (كما تظهر على معظم أقرص تليفيديو TeleVideo وكومباك Compaq، المهيئة في دوس DOS 2.x وأيضا على بعض أقراص دوس 3.1 إبسون Epson، وأوليفيتي Olivetti.

قفزة قصيرة SHORT JUMP		
0x90	0x??	0xEB

تعليمة القفزة القصيرة + إزاحة + تعليمة لا عملية

إزاحة."مسافة" يمثلها عدد مؤشر 1-بايت، المدى: $\pm 2^7$

قفزة قريبة NEAR JUMP		
0x??	0x??	0xE9

تعليمة القفزة القريبة (المباشرة)

(شفرة تشغيل بتبعها رقم إزاحة 16-بت مع إشارة) المدى: $\pm 2^{15}$

- هذه القفزة للتوافق مع الإصدارات السابقة. على الأقراص القابلة للإزالة. في MS-DOS, PC DOS, DR-DOS
- على الأقراص الثابتة DR-DOS يستخدم أيضا متتالية القفزة المبدلة بداية من تعليمة NOP، لكن أنظمة MS-DOS/PC DOS لا تفعل ذلك. (راجع الجدول Atari ST).

وجود أحد هذه الأنماط من شفرة التشغيل (مع تفحص قيمة واصف الوسيط عند الحيد 0x015) في دوس 3.3 والأحدث يخدم كمؤشر لوجود نوع من BPB (رغم أن الحجم الحقيقي لا يمكن تحديده من هدف القفزة لأن بعض قطاعات الاقلاع تتضمن بيانات محمل إقلاع خاصة تتبع BPB)، أما وحدات تخزين دوس 1 (وبعض وحدات تخزين دوس 3.0) فعليها العدوة إلى أسلوب دوس 1 للكشف عن البنية عبر بايت الوسيط في FAT (في القطاع المنطقي 1).

قفزة للتوافق مع الإصدارات السابقة		
0x??	0x??	0x69

قفزة (مبدلة) swapped JMPS		
0x??	0xEB	0x90

0x?? تعني أن أي قيمة من 8-بت يمكن أن تكون في هذا البايت.

0000h (0) | 3

هوية صانعي القطع الأصلية. هذه الهوية تشير إلى النظام المستخدم في تهيئة وحدة التخزين أو القرص. لكن نظام التشغيل لا يستخدمها بعد التهيئة. (هذه السلسلة الثمانية من محارف شفرة أسكي عادة محشوة بفراغات (بايت 0x20) أو ملحقة بأصفار إذا كان طول الاسم أقل من 8 محارف). رغم أنها رسميا محجوزة لفائدة صانعي القطع الأصلية OEM، أنظمة MS-DOS/PC DOS (منذ إصدار 3.1)، وويندوز SE/ME/95/98 و OS/2 تتفحص هذا الحقل لتقرير أية أجزاء الأخرى في سجل الإقلاع مطلوبة وكيفية ترجمتها. لهذا السبب تعيين هذه لصيقة إلى قيم زائفة أو اعتباطية يمكن أن يجعل تلك الأنظمة لا تتعرف على وحدة التخزين ويسبب تلف في البيانات عند الكتابة. أمثلة شائعة عن لصيقة اسم صانعي القطع الأصلية :

FreeDOS [sp]	mkdosfs [sp]	IBM [sp] [sp] 7.1	MSWIN4.1	MSDOS5.0	IBM [sp] [sp] 3.3

- إصدارات دوس الأقدم تستخدم أيضا في هذا الحقل لصيقة MSDOS5.1
- الأقراص المرنة المهيئة في لينكس قد تحمل في هذا الحقل القيمة mkdosfs.
- بعض الباعة/المنتجين يخزن أيضا معلومات الترخيص أو مفاتيح للنفاذ في هذا الحقل.
- بعض محملات الاقلاع تقوم بتعديلات أو ترفض تمرير التحكم إلى قطاع الاقلاع وفق قيم معينة تكتشفها هنا. (مثل NEWLDR عند الحيد 0x018).
- متتبع/متعقب وحدة التخزين في ويندوز SE/ME/95/98 سوف يستبدل لصيقة OEM بتواقيع "?????IHC" (و OGACIHC تعني شيكاغو Chicago وتشير إلى ويندوز 95) حتى على القرص الذي يبدو للقراءة-فقط (مثل :DIR A) إذا لم يكن الوسيط محمي من الكتابة.
- إذا أخذنا في الاعتبار بعض القيم المشروحة أعلاه، حسب بنية ومضمون الكتلة الفعلية BPB، قد يجعل هذا أنظمة MS-DOS/PC DOS، وويندوز SE/ME/95/98 و OS/2، لا تتعرف على الوسيط وتظهر رسائل أخطاء حتى وإن كان الوسيط سليم ويمكن قراءته بدون مشاكل في أنظمة التشغيل الأخرى.
- في حاسوب وانج، ذاكرة الاقلاع المقروءة فقط ROM boot تعتبر القرص قابل للإقلاع فقط إذا كانت المحارف الأربعة الأولى من لصيقة الاسم هي "Wang".
- في FAT32 EBPB إذا كان التوقيع عند حيد القطاع 42h هو 29h، وحقول القطاعات الإجمالية كلاهما كان 0، يمكن استخدام حقل نظام الملفات كحقل لحساب عدد القطاعات الإجمالي 64-بت واستخدام حقل لصيقة OEM كنوع نظام ملفات بديل بدلا من الحقل العادي عند الحيد 0x052.
- وبنفس الطريقة، إذا تم تعيين هذا الحقل إلى "EXFAT " سيشير إلى استعمال exFAT BPB الواقعة من عند حيد القطاع 40h إلى 77h.
- بينما وحدات تخزين NTFS تستخدم لصيقة "NTFS " لدلالة على NTFS BPB.

003h (3) | 8

معاملات الكتلة المعيارية والكتلة الممتدة

كتلة	معاملات الكتلة المعيارية BPB	معاملات الكتلة الممتدة EBPB	معاملات الكتلة الممتدة EBPB FAT32
أحجام	13، 19، 21، 25 بايت	32 أو 51 بايت	60 أو 79 بايت

الحجم والمحتوى يتفاوت بين أنظمة التشغيل والإصدارات، أنظر أدناه

00Bh (11) | متفاوت

متفاوت	متفاوت	**شفرة الاقلاع.**شفرة الاقلاع الخاصة بنظام الملفات ونظام التشغيل؛ غالبا تبدأ مباشرة خلف كتلة BPB[E]، لكن أحيانا توجد بيانات إضافية خاصة بمحمل الاقلاع مخزنة بين نهاية BPB[E] وبداية شفرة الاقلاع ؛ لذلك القفزة عند الحيد 001h لا يمكن الاعتماد عليها في استخراج بينة BPB[E] الدقيقة . بعض محملات إقلاع GPT (مثل BootDuet) مع كتلة DOS 3.31 BPB على الأقل،، تستخدم الحيود 0x1FA–0x1FD لتخزين 4 بايت العليا من القطاعات المخفية من أجل وحدات التخزين التي تقع خارج القطاعات الأولى 2^{32} -1. وبما أن هذا الموقع قد يتضمن شفرة أو بيانات أخرى في قطاعات الاقلاع أخرى، لا يجب أن يكتب إليه إن كانت الحيود 0x1F9–0x1FD لا تحتوي جميعها على أصفار.
1FDh (509)	1	**رقم القرص.** رقم القرص الفيزيائي (فقط في قطاعات إقلاع دوس 3.2 حتى 3.31). في نظام OS/2 1.0، ودوس 4.0، تم نقل هذا الحقل إلى حيد القطاع 024h (عند الحيد 019h في EBPB). ومنذ ذلك الحين، معظم قطاعات إقلاع مايكروسوفت و اي بي ام تحتفظ بقيم 00h عند الحيد 1FCh و 1FDh رغم أنها ليست جزء من التوقيع عند 1FEh. إذا كان هذا ينتمي إلى وحدة تخزين الاقلاع (قسم إقلاع)، يمكن ضبط MBR المحسن في DR-DOS 7.07. (أنظر NEWLDR عند 014h) كي يتم تحديث هذا الحقل ديناميكيا إلى قيمة DL زمن الاقلاع أو القيمة المخزنة في جدول الأقسام. هذا يسمح بإقلاع الأقراص البديلة، حتى عندما تتجاهل شفرة VBR قيمة DL.
1FEh (510)	2	توقيع قطاع الاقلاع (0x55 0xAA). هذا التوقيع يشير إلى شفرة إقلاع متوافقة مع أنظمة اي بي ام، معظم محملات الاقلاع الموجودة في MBR أو BIOS تتفحص هذا التوقيع قبل تمرير التنفيذ إلى شفرة الاقلاع في قطاع الاقلاع (لكن، أنظمة أخرى لا تفعل ذلك، مثل الأنظمة الأصلية من IBM PC ROM-BIOS). • هذا التوقيع لا يشير إلى نظام ملفات أو نظام تشغيل معين. • بما أن هذا التوقيع غير موجود على جميع الأقراص المهيئة بنظام FAT (مثلا، لا يوجد على وحدات تخزين DOS 1.x أو وحدات تخزين FAT التي لا تقبل الاقلاع في أنظمة x86)، أنظمة التشغيل ينبغي ألا تعتمد على وجود هذا التوقيع عند الولوج إلى وحدات التخزين (الإصدارات القديمة من MS-DOS/PC DOS قبل الإصدار 3.3 تتفحص هذا التوقيع، لكن الإصدارات الحديثة وكذلك نظام DR-DOS لا تفعل ذلك). • أدوات التهيئة لا يجب أن تكتب هذا التوقيع إذا كان قطاع الاقلاع المكتوب لا يتضمن محمل إقلاع افتراضي (روتين وهمي) واحد على الأقل متوافق مع أنظمة x86؛ على الأقل، كي يوقف CPU في حلقة لامتناهية (0xF4 0xEB 0xFD) أو يطلق INT 19h وشفرة العودة (أو القفزة) RETF (0xCD 0x19 0xCB). بأية حال شفرات التشغيل هذه لا ينبغي استخدمها عند حيد القطاع 000h، لأن نظام دوس يختبر شفرات التشغيل الأخرى كالتواقيع. • عدة أقراص مرنة في MSX-DOS 2 تستخدم EBh FEh 90h عند الحيد 000h، لوقف المعالج في المتكررة الحلقية tight loop بينما تحافظ على نمط شفرة تشغيل تفهمها أنظمة MS-DOS/PC DOS. • التوقيع يجب أن يكون عند الحيد الثابت 1FEh في قطاع 512 بايت أو أكثر. إذا كان حجم القطاع الفيزيائي أكبر، يمكن تكرار التوقيع عند نهاية القطاع الفيزيائي. • Atari ST سوف يفترض أن القرص هو أتاري 68000 Atari يقبل الاقلاع إذا كان تدقيق المجموع في قطاع الاقلاع فوق 256 كلمة نهوي-كبر يساوي 0x1234. إذا كانت شفرة محمل الاقلاع متوافقة مع IBM، يجب التأكد أن تدقيق المجموع فوق قطاع الاقلاع لا يتطابق عرضا مع تدقيق المجموع هذا. إن كان كذلك، يمكن تجنبه بتغيير بت غير مستخدم (مثلا، قبل أو بعد منطقة شفرة الاقلاع). • في حالات نادرة، لوحظ وجود توقيع معكوس 0x55 0xAA على صور القرص. هذا يمكن أن يكون بسبب تطبيق خاطئ في أداة التهيئة التي ترتكز على توثيق خاطئ، لكن يمكن أن يشير أيضا إلى ترتيب ثمانيات معكوس في صورة القرص، الذي يحدث عند النقل بين المنصات التي تستخدم طرق نهوية مختلفة. • قيم BPB وأنظمة الملفات FAT12، FAT16، FAT32 تستخدم فقط نظام نهوي-الصغير ولا يوجد أية تطبيقات للتنويعات تستخدم نظام نهوي-كبر.

قطاع إقلاع FAT12/16 (قطاع 0)

قطاع إقلاع وحدات تخزين FAT12 و FAT16 هو أول قطاع في المنطقة المحجوزة، بطول قطاع واحد، (512 بايت) ويتضمن التالي:

وصف	مثال	طول/بايت	رمز تذكري	إزاحة
تعليمة القفزة. تعليمة القفزة الغير مشروطة 3 بايت، (ستكون وفق نظام الملفات وموقع شفرة الاقلاع). عادة EB3C90h. هذه الشفرة من أجل القفز إلى موقع آخر في البرنامج وتنفيذ التعليمات هناك. ثلاثة بايت EB 3C 90 تفكك إلى JMP SHORT 3C NOP (قيمة 3C قد تكون مختلفة) : القفزة القصيرة، يجب أن تملك NOP حتى تشكل 3 بايت، وترجمتها: | قفزة قصيرة (إلى) | الحيد 3C | لا عملية | قيمة: EB — وظيفة: تعليمة قفزة قصيرة — قفزة قصيرة — بايت: 1 قيمة: 3C — وظيفة: إزاحة نسبية — SHORT JMP — بايت: 1 قيمة: 90 — وظيفة: تعليمة لا عملية NOP — بايت: 1	EB 3C 90	3	BS_jmpBoot	000h (0)
هوية صانعي القطع الأصلية،هوية صانعي القطع الأصلية OEM ID (أو هوية نظام التهيئة)، هذا الحقل يتضمن سلسلة محارف تشير إلى توقيع إصدارة النظام المستخدم في تهيئة وحدة التخزين. (القسم) عادة تكون MSWIN4.1. نظام التشغيل لا يستخدمها بعد التهيئة. رغم أن أنظمة مايكروسوفت لا تهتم بهذا الحقل بعض مشغلات FAT تفعل ذلك. وهذا سبب وجود سلسلة المحارف MSWIN4.1، كمحاولة للتجنب مشاكل التوافق. يمكن وضع نص أخر في هذا الحقل، لكن بعض مشغلات FAT لن تتعرف على وحدة التخزين. يمكن اعتبار هذا الحقل إشارة للنظام المستخدم في تهيئة وحدة التخزين. نظام التهيئة: ويندوز 95 OSR2 أو ويندوز 98 — هوية: MSWIN4.1 بعض المصادر تعتبر هذا الحقل جزء من كتلة BPB	MSWIN4.1	8	BS_OEMName	003h (3)

بداية معاملات الكتلة FAT12/16 BPB

أصل المعاملات الثمانية التالية من كتلة: DOS 2.0 BPB (FAT12)

عدد بايتات في كل قطاع. حجم القطاع على الوسيط الفيزيائي، في العادة يكون 512 بايت. ويمكن أيضا أن يكون بإحدى القيم التالية: عشري: 512 — ست عشري: 200h — ترتيب بايت: 00 02 1024 — 400h — 00 04 2048 — 800h — 00 08 4096 — 1000h — 00 10 للتوافق يجب أن تكون دائما 512 بايت، غالبا الأنظمة لن تحاول فحص هذه القيمة لأن معظم شفرة FAT مصممة لاستخدام 512 بايت في كل قطاع. ورغم أن أنظمة مايكروسوفت تدعم القيم الأخرى، لكن لا ينصح باستخدامها.	00 02	2	BPB_BytsPerSec	00Bh (11)
عدد القطاعات المنطقية لكل عنقود. حجم العنقود يجب أن يكون قوة العدد 2 أكبر من 0 ؛ عادة تكون بإحدى القيم التالية: عشري: 1 قطاع — ست عشري: 1h — بايت: 512 — كيلوبايت: 0.5 2 قطاع — 2h — 1024 — 1 4 قطاع — 4h — 2048 — 2 8 قطاع — 8h — 4096 — 4 16 قطاع — 10h — 8192 — 8 32 قطاع — 20h — 16384 — 16 64 قطاع — 40h — 32768 — 32 128 قطاع — 80h — 65536 — 64 نظرا لأن FAT16 مقيد بعدد العناقيد التي يستطيع تعقبها، (تصل إلى 65524)، دعم وحدات التخزين الكبيرة سيكون بزيادة عدد القطاعات في كل عنقود. لكن الحجم المبدئي للعنقود في وحدة التخزين سوف يعتمد كليا على حجم وحدة التخزين (ونظام الملفات المستخدم). لا ينبغي استخدام حجم عنقود أكبر من 32 كيلوبايت (32768 بايت). بمعنى آخر، القيم التي تجعل حجم العنقود أكبر من 32 كيلوبايت قد لا تعمل بالشكل الصحيح وليست مدعومة في الكثير من الأنظمة؛ لا تحاول تعيين إحداها. رغم أن بعض إصدارات أنظمة التشغيل تسمح بالقيمة 64 كيلوبايت لكل عنقود. عدة تطبيقات لن تعمل على مثل وحدات التخزين FAT. هذه.	04	1	BPB_SecPerClus	00Dh (13)
القطاعات المحجوزة (1 قطاع في FAT12/16، و 32 قطاع في FAT32). عدد القطاعات المحجوزة (من أجل سجل الاقلاع) بداية من أول قطاع في وحدة التخزين إلى بداية أول نسخة FAT ؛ يجب أن تكون دائما 1 على وحدات تخزين FAT12 و FAT16. القيمة 0 لا تصلح أبدا في هذا الحقل لأن المنطقة المحجوزة دائما تتضمن قطاع الاقلاع، هذه القيمة تستخدم لحساب موقع القطاع الأول الذي يتضمن نسخة FAT. بمعنى آخر، هذا الحقل يمثل عدد القطاعات التي تسبق بداية أول نسخة من FAT، وتشمل قطاع الاقلاع نفسه. عدة شفرات FAT مصممة للعمل مع قطاع واحد، محجوز من أجل وحدات تخزين FAT12 و FAT16. لذلك تلك الشفرات لن تحاول التأكد من وجود القيمة 1 في هذا الحقل. هذه القيمة في FAT32 عادة هي 20h (أي 32).	01 00	2	BPB_RsvdSecCnt	00Eh (14)

				إذا كانت القيمة أكبر من 1، فذلك يعني أن شفرة الاقلاع كبيرة جدا ولا تتنسب مع الحيز المخصص في قطاع إقلاع القسم.
010h (16)	BPB_NumFATs	1	02	**عدد نسخ FAT**. (عادة تكون 2.). هذا الحقل الموجود في كافة أنواع وحدات تخزين FAT. يمثل عدد نسخ FAT المخزن على القرص. عادة، قيمة هذا الحقل هي 2 (ويمكن أن تكون 1 أو أكثر، رغم أن بعض الأنظمة لا تدعم أكثر من 2). هذه النسخ ستكون متعاقبة على القرص، أي النسخة FAT2 تأتي مباشرة بعد النسخة الأولى FAT1. وينبغي دائما أن تكون هناك نسخة واحدة على الأقل حاضرة. جميع مشغلات أنظمة ملفات مايكروسوفت تدعم أيضا القيم الأخرى في هذا الحقل. في الوسائط الأخرى، مثل بطاقات ذاكرة فلاش، هذه الوظيفة لا فائدة منها في حماية البيانات، وتستخدم القيمة 1 للاستفادة من المساحة التي كانت ستشغلها النسخة FAT2. لكن بعض مشغلات FAT لا تتعرف على مثل وحدات التخزين هذه.
011h (17)	BPB_RootEntCnt	2	00 02	**عدد مدخلات الجذر** (الممكنة أو الأقصى). عادة تكون 512.. إجمالي عدد المدخلات 32-بات الخاصة بأسماء الملفات والمجلد التي يمكن تخزينها في المجلد الجذر (ذو المركز والحجم الثابتين) في وحدة التخزين.على القرص العادي للتوافق قيمة هذا الحقل ستكون 512 . علما أن إحدى هذه المدخلات سوف تستخدم كلصيقة اسم وحدة التخزين، والملفات والمجلدات التي بأسماء ملفات طويلة تستخدم عدة مدخلات في كل ملف. هذا يعني أن العدد الأكبر لمدخلات الملف والمجلد في الدليل الجذر هو 511، لكن إذا استخدمت أسماء الملفات الطويلة، سوف تنفد المدخلات قبل أن تصل إلى ذلك الرقم (أي تستهلكها أسماء الملفات طويلة). إذا كان الدليل الجذر غير ثابت هذه القيمة ستكون صفر. لهذا السبب جميع أقراص FAT32 يجب أن تتضمن القيمة صفر في هذا الحقل، هذا يعني أن الدليل الجذر سيكون بطول كيفي. ما عدا ذلك هذا الحقل عادة يتضمن القيمة 512. وكل مدخلة دليل تستهلك 32 بايت. لكن لتجنب هدر المساحة، يجب أن يقبل هذا الحقل القسمة على حقل عدد بايتات لكل قطاع. عدد مدخلات الجذر * 32 / عدد بايتات لكل قطاع . مثال: 224 مدخلات * 32 بايت / 512 بايت = 14 قطاع هي طول جدول الدليل
013h (19)	BPB_TotSec16	2	00 00	**عدد القطاعات الإجمالي**. في الأقراص الصغرى فقط (إذا كانت وحدة التخزين أقل من 32 ميغابايت). إجمالي عدد القطاعات 16-بت، تشمل القطاعات التي تحتلها المناطق الأربعة ويتكون منها نظام ملفات على وحدات التخزين. • إذا كانت وحدات التخزين أصغر من 65.536 قطاع، هذا الحقل سيتضمن العدد الإجمالي للقطاعات، وحقل العدد الإجمالي الأكبر للقطاعات 32-بت سيكون 0، وهوية النظام في MBR ستكون 01h إذا كانت وحدات تخزين FAT12 أو 04h إذا كانت FAT16. • إذا كانت وحدة التخزين 65.536 قطاع أو أكبر هذا الحقل سيكون 0 ، وحقل العدد الإجمالي الأكبر للقطاعات 32-بت سوف يتضمن العدد الفعلي للقطاعات. هوية النظام في MBR ستكون 06h. (راجع كتيب: نوع القسم) ينبغي أن يتضمن نفس قيمة أو أقل الحقل المقابل في جدول الأقسام. إذا كانت القيم غير متساوية، ستستخدم القيمة الأصغر في احدهما. هذا الحقل ينبغي أن يكون كبير بما فيه الكفاية حتى يتضمن على الأقل القطاعات المحجوزة، وجميع نسخ FAT، والدليل الجذر، إن وجد.
015h (21)	BPB_Media	1	F8	**واصف الوسيط**. بايت واصف الوسيط ؛ عادة يكون F8h على جميع الوسائط الثابتة، و F0h على معظم الوسائط التي تقبل الفصل. هذا البايت يقدم معلومات عن الوسيط المستخدم. الجدول التالي يعرض بعض قيم واصف الوسيط المعترف بها مع وسائطها (في قطاع إقلاع FAT). لحظ أن بايت واصف الوسيط يمكن أن يرتبط بأكثر من سعة للقرص. هذه نفس القيمة التي يجب أن تكون في البايت السفلي في المدخلة الأولى من FAT1

	دوس	عدد المسارات	رؤوس	قطاعات	نوع القرص	سعة	بايت
ذو وجهين		80	2	36	3.5-بوصة	2.88 ميغابايت	F0
ذو وجهين	3.3	80	2	18	3.5-بوصة	1.44 ميغابايت	F0
	2.0	قرص مثبت				؟	F8
ذو وجهين	3.2	80	2	9	3.5-بوصة	720 كيلوبايت	F9
ذو وجهين	3.0	80	2	15	5.25-بوصة	1.2 ميغابايت	F9
وجه واحد	2.0	80	1	8 أو 9 ؟	كلاهما	320 كيلوبايت	FA
		من أجل أقراص الذاكرة RAM disk				؟	FA
ذو وجهين		80	2	8 أو 9 ؟	كلاهما	640 كيلوبايت	FB
وجه واحد	2.0	40	1	9	5.25-بوصة	180 كيلوبايت	FC
؟		77	؟	؟	8-بوصة	؟ كيلوبايت	FC
ذو وجهين	2.0	40	2	9	5.25-بوصة	360 كيلوبايت	FD
ذو وجهين	2.0	77	2	26	8-بوصة	500 كيلوبايت	FD
وجه واحد	1.0	40	1	8	5.25-بوصة	160 كيلوبايت	FE
ذو وجهين	2.0	77	2	8	8-بوصة	1.2 ميغابايت	FE
وجه واحد	1.0	77	1	26	8-بوصة	250 كيلوبايت	FE
ذو وجهين	1.1	40	2	8	5.25-بوصة	320 كيلوبايت	FF

هذه القيم تعود إلى أقراص مايكروسوفت دوس القديمة (FAT12/16) وتقريبا لم تعد تستخدم.

رقم المسار ورقم الأسطوانة ورقم الرأس و رقم الجانب جميعها يرتكز على الصفر. على سبيل المثال، الأقراص المرنة 5.25 بوصة حجم 360 كيلوبايت

تملك 40 مسار من 0 إلى 39. كذلك الحال مع أرقام الرؤوس والجوانب. مثلا، نفس الأقراص المرنة 5.25 بوصة تملك جوانب من 0 إلى 1 (تقابل الرأس 0 و 1). لكن القطاعات ترتكز على الواحد. مثلا، الأقراص المرنة 5.25 بوصة حجم 360 كيلوبايت تملك قطاعات من 1 إلى 9.				
عدد القطاعات لكل FAT. (قطاعات 16-بت). عدد القطاعات المشغولة من كل نسخة FAT على وحدة التخزين. الحاسوب سوف يستخدم هذا العدد مع عدد نسخ FATs وعدد القطاعات المحجوزة لتحديد بداية الدليل الجذر. الحاسوب يستطيع أيضا تحديد بداية منطقة بيانات المستخدم على وحدة التخزين وفقا لعدد المدخلات في الدليل الجذر. (512) • في قرص FAT32 هذا لا يستخدم وسيكون 0، والقيمة الفعلية تكون في حقل عدد القطاعات لكل نسخة FAT، 32 بت. • في مايكروسوفت، محمل إقلاع FAT32 لن يعمل مع أقراص FAT32 إذا كان الحقل ليس صفر، وكذلك محملات إقلاع FAT12/16 لن تعمل مع الأقراص إذا كان الحقل صفر. لذلك القيمة يجب أن تكون مناسبة لهذا الحقل، كما هو حال حقل العدد الإجمالي الأصغر للقطاعات 16-بت.	CD 00	2	BPB_FATSz16	016h (22)

أصل المعاملات الثلاثة التالية، الكتلتان: DOS 3.0 BPB (FAT12, FAT16), DOS 3.31 BPB (FAT12, FAT16, FAT16B)

الحقلان المقابلان لحقل عدد القطاعات في كل مسار، وحقل عدد الرؤوس متماثل في DOS 3.0 BPB و DOS 3.31 BPB، لكن حقل القطاعات المخفية غير متوافق في DOS 3.0 BPB و DOS 3.31 BPB

عدد القطاعات لكل مسار. يمثل عدد القطاعات لكل مسار (وعدد القطاعات لكل رأس). عادة يكون 63 على القرص الثابت. (المرئي للنداء المقاطعة 13h). هذه القيم جزء من قياسات القرص الظاهرية المستخدمة عند تهيئة القرص (تهيئة منخفضة المستوى) عدد القطاعات لكل رأس هو عدد القطاعات المجتمعة تحت رأس. • عدد الرؤوس لكل أسطوانة (عند 0x1A) يعكس عدد الأسطوانات لكل رأس. إذا كان هذا القسم يستخدم عنونة الكتل الفيزيائية CHS، يجب أن تكون هذه القيم نفس القيم العائدة من نظام BIOS. إذا كانت مختلفة، فذلك يعني أن إعدادات القرص خاطئة والقسم لا يصلح للاستعمال. • في مايكروسوفت، محمل الاقلاع يغير جدول معاملات القرص المرن DPT في نظام BIOS عن طريق تعيين حقل عدد القطاعات لكل مسار لهذه البنية إلى عدد القطاعات لكل رأس كما تقرا من قرص الاقلاع. • القيم في هذه الحقول لا تهم نظام عنونة الكتل المنطقية LBA. • الرأس head = الجانب side : أحيانا كلمة رأس تستبدل بكلمة جانب، كما في الوسيط القابل للفصل مثل القرص المرن. • مسار track=أسطوانة cylinder في المرن ذو وجه واحد/وجهين، يستخدم عادة المسار. و في حالة أكثر من رأسين الأسطوانة، السعة القصوى للوسيط = حجم القطاع * عدد القطاعات لكل مسار * الأسطوانات * الرؤوس	3F 00	2	BPB_SecPerTrk	018h (24)
عدد الرؤوس. عدد الرؤوس/الجوانب (لكل أسطوانة) (قراءة/كتابة) عادة تكون 255 على القرص الثابت. في القرص المرن ذو وجهين ستكون 2. هذه القيم جزء من قياسات القرص الظاهرية المستخدمة عند تهيئة القرص (تهيئة منخفضة المستوى).	04 00	2	BPB_NumHeads	01Ah (26)
القطاعات المخفية. هذا هو عدد القطاعات (المخفية) على القرص الفيزيائي التي تسبق بداية القسم الذي يتضمن وحدة التخزين، قبل قطاع الاقلاع نفسه. عادة تكون 63 لأول وحدة تخزين. • هذا الحقل له علاقة فقط بالوسيط المرئي للنداء المقاطعة 13h. • هذا الحقل يجب أن يكون دائما 0 في الوسيط الغير مقسم (أي بدون أقسام). • هذا الحقل يستخدم أثناء عملية الاقلاع لحساب الحيد المطلق إلى الدليل الجذر ومناطق السانات. يمكن اعتباره عدد القطاعات بين بداية القسم وجدول الأقسام نفسه. • في الإصدارات القديمة من دوس، 2 بايت العليا (high word)، عادة تتضمن ما يدعى garbage. • هذا الحقل يجب أن يكون نفس عدد القطاعات التي تسبق القسم في جدول الأقسام. إذا كانت القطاعات المخفية ليست مثل التي في حقل القطاع النسبي في جدول الأقسام، يعتبر قطاع الاقلاع فاسد والقسم لا يصلح للاستعمال. garbage في الحوسبة، تعني بيانات خاطئة، بلا معنى، أو غير مرغوبة. أو إدخال غير مناسب أو خاطئ ؛ البيانات بلا فائدة أو لم تعد مطلوبة.	3F 00 00 00	4	BPB_HiddSec	01Ch (28)

أصل المعامل التالي كتلة: DOS 3.2 BPB (FAT12, FAT16)

العدد الإجمالي للقطاعات غير متوافق بين DOS 3.2 BPB و DOS 3.31 BPB

عدد القطاعات الإجمالي. في الأقراص الكبرى (إذا كانت وحدة التخزين أكبر من 32 ميغابايت). إجمالي عدد القطاعات 32-بت، تشمل القطاعات التي تحتلها المناطق الأربعة ويتكون منها نظام الملفات على وحدات التخزين. • إذا كانت وحدة التخزين بحجم 65.536 قطاع أو أكثر هذا الحقل سوف يتضمن العدد الإجمالي للقطاعات، وحقل العدد الاجمالي الأصغر للقطاعات 16-بت سيكون 0. وهوية النظام في MBR ستكون 06h). • إذا كانت وحدات التخزين أصغر من 65.536 قطاع، هذا الحقل يجب أن يكون 0، وحقل العدد الإجمالي الأصغر للقطاعات 16-بت سوف يتضمن التعداد. وهوية النظام في MBR سوف تكون 01h إذا كانت وحدات تخزين FAT12 و 04h إذا كانت FAT16.	C1 32 03 00	4	BPB_TotSec32	020h (32)

بعد معاملات الكتلة المعيارية FAT12/16 BPB تأتي إما حقول الكتلة الممتدة FAT12/16 EBPB (استخدمت لأول مرة في الأقراص المهيئة في أنظمة DOS 4.0، OS/2 1.0، PC DOS 3.4) أو شفرة للآلة (شفرة الاقلاع)، وتتضمن معلومات تستخدم فقط في وحدة تخزين FAT12 و FAT16. الحقول التالية سوف تختلف عن نظيرتها في FAT32 بداية من الحيد 024h.

بنية EBPB في وحدات تخزين FAT12, FAT16, FAT16B, HPFS			ملاحظة	
اسم الحقل	بايت	إزاحة	DOS 3.4 EBPB (FAT12, FAT16, FAT16B)	DOS 4.0 EBPB (FAT12, FAT16, FAT16B, HPFS)
رقم القرص الفيزيائي	1	024h (36)		رقم القرص مشابه DOS 3.4 EBPB
الرأس الحالي (محجوزة ؛ أعلام ويندوز أن تي)	1	025h (37)		
توقيع إقلاع ممتد	1	026h (38)	التوقيع 0x28 يعرف بـ "4.0" ومشابه لكتلة DOS 4.0 EBPB و NTFS EBPB	التوقيع 0x29 يعرف بـ "4.1") ومشابه لكتلة DOS 3.4 EBPB و NTFS EBPB
الرقم التسلسلي لوحدة التخزين	4	027h (39)		رقم تسلسلي مشابه DOS 3.4 EBPB
لصيقة اسم وحدة التخزين	11	02Bh (43)		
نوع نظام الملفات	8	036h (54)		

رقم القرص. رقم القرص المنطقي في دوس المرتبط برقم القرص الفيزيائي في BIOS المستخدم مع نداء المقاطعة INT 13h. ترقيم الأقراص المرنة يبدأ من 0x00 (دائما تأخذ "A" أو "B")، بينما تبدأ من 0x80. في الأقراص الثابتة (80h = قسم أولي)، بغض النظر عن عدد الأقراص الثابتة الموجودة، لأن هذه القيمة لها علاقة فقط بقرص الاقلاع المحدد في نظام BIOS. وعادة يتم تعيين هذه القيمة قبل إصدار نداء المقاطعة INT 13 BIOS حتى يتم تحديد القرص الذي سيكون النفاذ إليه.	80	1	BS_DrvNum	024h (36)
محجوزة.قيمة هذا الحقل ستكون دائما 0. عند تهيئة وحدات تخزين FAT. لكن ويندوز أن تي يستخدم هذا البايت لتخزين اثنان من أعلام CHKDSK. • بت 0 (ترتيب منخفض) علم وحدة التخزين كثيرة الأخطاء، ويشير لضرورة تشغيل الفحص الألي في الاقلاع التالي. • بت 1 (ترتيب منخفض) علم أخطاء الإدخال/الأخراج I/O، يشير لضرورة عمل تفحص دقيق للسطح في إقلاع تالي. (البتات 7-2 ستكون دائما خالية). وينبغي لأدوات التهيئة تعين الحقل إلى 0. الرأس الحالي: في الأصل، كانت تستخدم لتخزين الأسطوانة/المسار الذي (يتضمن) يقع عليه قطاع الاقلاع، لكن القيمة المخزنة على القرص حاليا لا تستخدم على هذا النحو. لمعلومات أكثر راجع كتلة معاملات BPB في الموسوعة الحرة.	00	1	BS_Reserved1	025h (37)
توقيع إقلاع ممتد يجب أن تكون 0x28 أو 0x29 حتى تتعرف عليه أنظمة ويندوز أن تي. إذا كانت القيمة هي 29h، سوف تشير إلى وجود الحقول الثلاثة التالية. وإلا، ستكون 00h. حقل / حجم / إزاحة الرقم التسلسلي لوحدة التخزين / 4 بايت / 027h (39) لصيقة اسم وحدة التخزين / 11 بايت / 02Bh (43) نوع نظام الملفات / 8 بايت / 036h (54)	29	1	BS_BootSig	026h (38)
الرقم التسلسلي لوحدة التخزين 32-بت. رقم تسلسلي عشوائي لكن فريد ومخزن في ترتيب معكوس، يستخدم مع لصيقة اسم وحدة التخزين. لتعقب وحدة التخزين على الوسيط القابل للإزالة (يسمح لمشغلات FAT معرفة هوية القرص). عادة القيمة تولد باستخدام التاريخ والوقت زمن تهيئة وحدة التخزين. علما أنه لا يمكن استرجاع (استخراج) التاريخ والوقت بواسطة الرقم التسلسلي. مثال على حسابها ثم تحويلها إلى النظام الست عشري وقت / تاريخ / رقم تسلسلي مساء 2:22:32.50 / أكتوبر 10-03-2001 / 35,2A,E7,15 أول بايت / ثاني بايت / 2 بايت الأخيرة ملي ثانية + الأيام / الأشهر + الثواني / (الساعات [إذا كان المساء + 12] × 256) + الدقائق + السنوات 50 + 3 = 53 / 10 + 32 = 42 / (2 + 12 = 14 * 256 = 3584) + 22 + 2001 = 5607 35 -> = 53 2A -> = 42 E7,15 -> 15,e7 -> = 5607	ED 15 77 38	4	BS_VolID	027h (39)
لصيقة اسم وحدة التخزين. لصيقة اسم وحدة التخزين (في القسم) يجب أن تتطابق مع مدخلة لصيقة اسم وحدة التخزين (في الدليل الجذر) المخزنة الآن كملف خاص. إذا كانت المدخلة غير موجودة، يجب أن يتضمن هذا الحقل القيمة NO NAME. دوس يضع هذه لصيقة عند تهيئة القرص ويجددها مع مدخلة لصيقة وحدة التخزين في الدليل الجذر عند إنشائها أو تغييرها مثلا بـ Label.	MSDOS710	11	BS_VolLab	02Bh (43)
نوع نظام الملفات. وثائق مايكروسوفت الخاصة بنظام ملفات FAT16 تقول أن هذا الحقل غير مستخدم من قبل ويندوز أكس بي. وهذا الحقل عبارة عن سلسلة محارف، محشوة بفراغات. هذا الحقل لغرض معلوماتي/اعلامي فقط. عند تهيئة القرص يجب أن يكون بإحدى السلاسل التالية FAT16 ␣␣␣␣ / FAT12 ␣␣␣␣ / FAT ␣␣␣␣␣␣ يجب تعيين قيمة هذا الحقل وفق نوع نظام الملفات FAT المستخدم، رغم أن هذا الحقل:	FAT16	8	BS_FilSysType	036h (54)

لا يحدد نوع نظام الملفات، وليس جزء من معاملات كتلة BPB/EBPB ! وقد يكون تعيين هذا الحقل خاطئ أو غير موجود، ونظام التشغيل لا يستخدم هذا الحقل بعد التهيئة. ومع ذلك كله، بعض مشغلات نظام ملفات FAT من خارج مايكروسوفت سوف تتفحص هذا الحقل.				

نهاية معاملات الكتلة الممتدة FAT12/16 EBPB وبداية شفرة الاقلاع وتوقيع الاقلاع

شفرة الاقلاع. بداية الجزء الذي بدون تهيئة من قطاع الاقلاع. والذي يتضمن بيانات وشفرة يتم تنفيذها عند إقلاع القرص. في وحدة التخزين القابلة للإقلاع، المنطقة التي تلي كتلة EBPB عبارة عن شفرة إقلاع تنفيذية. مسؤولة عن تنفيذ كل ما هو مطلوب من أجل استمرار عملية الاقلاع. هذه الشفرة سوف تكون مختلفة بين أنظمة التشغيل وإصداراتها، مثلا في مايكروسوفت دوس، شفرة الاقلاع سوف تحدد موقع ملف IO.SYS في نظام الملفات، وتحمل جزء منه في الذاكرة ثم تقفز إلى مدخلة مخصصة في IO.SYS. في أنظمة ويندوز أن تي، هذه الشفرة تحدد موقع الملف NTLDR، ثم تحميله في الذاكرة، ثم تنقل عملية التنفيذ إلى ذلك الملف. أي عمل شفرة الاقلاع سيكون مختلف بين أنظمة التشغيل. أيضا الأقراص المرنة التي لا تقبل الاقلاع تملك شفرة تنفيذية في هذه المنطقة. الشفرة ضرورية لعرض رسائل الأخطاء المألوفة، "Non-system disk أو disk error" في معظم الأقراص المرنة المعيارية المهيئة في م.س-دوس بدون استخدام خيار "system".	متفاوت	448	BS_BootCode	03Eh (62)
توقيع قطاع الاقلاع. (يجب أن يكون دائما 55AAh). هذه الشفرة تخبر نظام BIOS أن هذا القطاع قطاع تنفيذي (يقبل الاقلاع). التطبيقات الأخرى تستخدم هذا التوقيع لتحقق من تحميل القطاع الصحيح. ومهما كانت قيمة عدد بايتات لكل قطاع. هذا التوقيع يجب أن يكون دائما عند الحيد 1FEh (أي عند 510-511). إذا كان القطاع بحجم 512 بايت فهذه تعني أيضا نهاية القطاع.	55 AA	2	BS_Signature	1FEh (510)

بنية قطاع إقلاع FAT12/16 (الجزء العلوي)

- تعليمة القفزة، الغير مشروطة (3 بايت)
- هوية صانعي القطع الأصلية (النظام المستخدم في التهيئة) (8 بايت)

منطقة معاملات BPB

- عدد بايتات في كل قطاع (2 بايت)
- عدد القطاعات لكل عنقود (1 بايت)
- عدد القطاعات المحجوزة بداية من أول قطاع في وحدة التخزين (2 بايت)
- عدد نسخ FAT في نظام الملفات (1 بايت)
- عدد مدخلات الجذر المحتملة (2 بايت)
- العدد الإجمالي الأصغر للقطاعات 16-بت (2 بايت)
- بايت واصف الوسيط (1 بايت)
- عدد القطاعات في كل نسخة FAT (2 بايت)
- عدد القطاعات لكل مسار وعدد القطاعات لكل رأس(2 بايت)
- عدد الرؤوس لكل أسطوانة (2 بايت)
- عدد القطاعات المخفية التي تسبق القسم (4 بايت)
- العدد الأكبر الإجمالي للقطاعات 32-بت (الجديد) (4 بايت)

منطقة معاملات EBPB

- رقم القرص الفيزيائي (1 بايت)
- محجوزة (1 بايت)
- توقيع إقلاع ممتد (1 بايت)
- الرقم التسلسلي لوحدة التخزين (4 بايت)
- لصيقة اسم وحدة التخزين (في القسم) (11 بايت)
- نوع نظام الملفات (8 بايت)

منطقة الشفرة

- شفرة إقلاع تنفيذية. سوف تكون مختلفة بين أنظمة التشغيل

بنية قطاع إقلاع FAT12/16 (الجزء السفلي)

رسائل الأخطاء وأسماء ملفات النظام الثلاثة

- مواقع البيانات (4 بايت)
- رسائل الأخطاء (84 بايت !)
- أسماء ملفات النظام الثلاثة (11 بايت)
- توقيع قطاع الاقلاع ؛ سيكون دائما 55AAh (إذا كان القطاع سليم !).
- أسماء ملفات النظام الثلاثة (11 بايت)
- مواقع البيانات (3 بايت)
- أسماء ملفات النظام الثلاثة (11 بايت)

(القطاع المطلق) CHS 0-1-**1**, LBA **63**

```
      00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F  0123456789ABCDEF
0000: EB 3C 90 4D 53 57 49 4E 34 2E 31 00 02 04 01 00  [.<.MSWIN4.1.....]
0010: 02 00 02 00 00 F8 CD 00 3F 00 04 00 3F 00 00 00  [........?...?...]
0020: 21 32 03 00 80 00 29 ED 15 77 38 4D 53 44 4F 53  [.2....)..w8MSDOS]
0030: 37 31 30 20 20 20 46 41 54 31 36 20 20 20 33 C9  [710   FAT16   3.]
0040: 8E D1 BC FC 7B 16 07 BD 78 00 C5 76 00 1E 56 16  [....{...x..v..V.]
0050: 55 BF 22 05 89 7E 00 89 4E 02 B1 0B FC F3 A4 06  [U."..~..N.......]
0060: 1F BD 00 7C C6 45 FE 0F 38 4E 24 7D 20 8B C1 99  [...|.E..8N$} ...]
0070: E8 7E 01 83 EB 3A 66 A1 1C 7C 66 3B 07 8A 57 FC  [.~...:f..|f;..W.]
0080: 75 06 80 CA 02 88 56 02 80 C3 10 73 ED 33 C9 FE  [u.....V....s.3..]
0090: 06 D8 7D 8A 46 10 98 F7 66 16 03 46 1C 13 56 1E  [..}.F...f..F..V.]
00A0: 03 46 0E 13 D1 8B 76 11 60 89 46 FC 89 56 FE B8  [.F....v.`.F..V..]
00B0: 20 00 F7 E6 8B 5E 0B 03 C3 48 F7 F3 01 46 FC 11  [ ....^...H...F..]
00C0: 4E FE 61 BF 00 07 E8 28 01 72 3E 38 2D 74 17 60  [N.a....(.r>8-t.`]
00D0: B1 0B BE D8 7D F3 A6 61 74 3D 4E 74 09 83 C7 20  [....}..at=Nt... ]
00E0: 3B FB 72 E7 EB DD FE 0E D8 7D 7B A7 BE 7F 7D AC  [;.r......}{...}.]
00F0: 98 03 F0 AC 98 40 74 0C 48 74 13 B4 0E BB 07 00  [.....@t.Ht......]
0100: CD 10 EB EF BE 82 7D EB E6 BE 80 7D EB E1 CD 16  [......}....}....]
0110: 5E 1F 66 8F 04 CD 19 BE 81 7D 8B 7D 1A 8D 45 FE  [^.f......}.}..E.]
0120: 8A 4E 0D F7 E1 03 46 FC 13 56 FE B1 04 E8 C2 00  [.N....F..V......]
0130: 72 D7 EA 00 02 70 00 52 50 06 53 6A 01 6A 10 91  [r....p.RP.Sj.j..]
0140: 8B 46 18 A2 26 05 96 92 33 D2 F7 F6 91 F7 F6 42  [.F..&...3......B]
0150: 87 CA F7 76 1A 8A F2 8A E8 C0 CC 02 0A CC B8 01  [...v............]
0160: 02 80 7E 02 0E 75 04 B4 42 8B F4 8A 56 24 CD 13  [..~..u..B...V$..]
0170: 61 61 72 0A 40 75 01 42 03 5E 0B 49 75 77 C3 03  [aar.@u.B.^.Iuw..]
0180: 18 01 27 0D 0A 49 6E 76 61 6C 69 64 20 73 79 73  [..'..Invalid sys]
0190: 74 65 6D 20 64 69 73 6B FF 0D 0A 44 69 73 6B 20  [tem disk...Disk ]
01A0: 49 2F 4F 20 65 72 72 6F 72 FF 0D 0A 52 65 70 6C  [I/O error...Repl]
01B0: 61 63 65 20 74 68 65 20 64 69 73 6B 2C 20 61 6E  [ace the disk, an]
01C0: 64 20 74 68 65 6E 20 70 72 65 73 73 20 61 6E 79  [d then press any]
01D0: 20 6B 65 79 0D 0A 00 00 49 4F 20 20 20 20 20 20  [ key....IO      ]
01E0: 53 59 53 4D 53 44 4F 53 20 20 20 53 59 53 7F 01  [SYSMSDOS   SYS..]
01F0: 00 41 BB 00 07 60 66 6A 00 E9 3B FF 00 00 55 AA  [.A...`fj..;...U.]
      00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F
```

قطاع إقلاع وحدة التخزين FAT16، ويندوز 95. كل سطر 16 بايت (32 سطر). عند العنوان: 63 LBA (1-1-0 CHS)

بايت	شفرة أسكي (محرف) / بايت ست عشري			
1	CRLF	Carriage Return	0Dh (13)	محرف رجوع إلى السطر، مرجع إلى السطر
1		Line Feed	0Ah (10)	محرف تزويد سطر
1	(sz)		00h (00)	نهاية صفرية (سلسلة + بايت صفر)
1			FFh (255)	بايت يشير إلى الشفرة التي سيتم تفحصها

كل سطر رسالة خطأ يبدأ بـ 2 بايت 0Dh و 0Ah. وينتهي بايت FFh وينتهي أخيرا بايت 00h

سجل إقلاع FAT32

في وحدة تخزين FAT32، سجل إقلاع القسم بطول 3 قطاعات، رغم أن هناك 32 قطاع محجوزة من أجل سجلات الاقلاع الاعتيادية والاحتياطية في بداية وحدة التخزين. سجل إقلاع القسم الاعتيادي يقع في القطاعات المنطقية 0، 1، 2، (أي قطاعات القرص المطلقة 63، 64، 65) والنسخة الاحتياطية من سجل الاقلاع تقع في القطاعات 6، 7، و 8. (القطاعات المطلقة 69، 70، 71) وجميعها تنشأ عند تهيئة وحدة التخزين ولا تتغير طوال فترة الاستعمال العادي. هذا بالنسبة للقسم الأول على القرص (أو القسم الوحيد على القرص). لكن القطاعات المطلقة يمكن أيضا أن تتواجد في مكان أخر إذا كان القرص يملك أقسام لأنظمة أخرى.

في ويندوز 98 سجل الاقلاع يحتل أول 6 قطاعات من وحدة التخزين، يستخدم منها فقط القطاعات الثالثة الأولى. النسخة الاحتياطية من سجل الاقلاع ستكون في الستة قطاعات اللاحقة. ثم يأتي بعدها مباشرة 20 قطاع أخر تتضمن فقط قيم أصفار لكنها تبقى جزء من المنطقة المحجوزة (أي 32 قطاع).

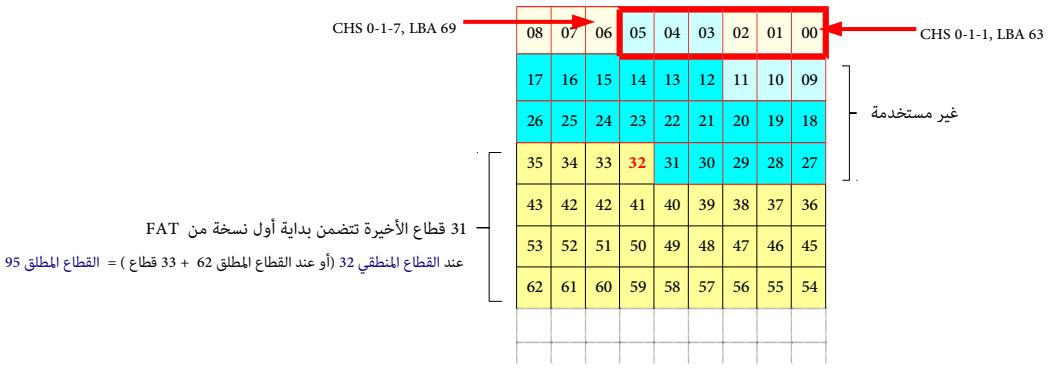
أول قطاع من كل سجل إقلاع يتضمن جزء من شفرة الاقلاع ومعاملات BPB/EBPB. القطاع الثاني يشغله قطاع معلومات نظام الملفات FAT32. والقطاع الثالث يتضمن بقية شفرة الاقلاع.

الاختلاف الوحيد بين نسختي سجل الاقلاع (الاعتيادية والاحتياطية) سيكون في قطاع معلومات نظام الملفات، وتحديدا في قيمة حقل حساب العناقيد الحرة (4-بايت).

في كل قطاع من الثلاثة قطاعات، التوقيع 0xAA55 يحتل 2 بايت الأخيرة في القطاع 512 بايت، أيضا في القطاع الأكبر سيظل التوقيع عند موقع البايت 511 و 512.

بنية قطاع إقلاع وحدة تخزين FAT32 تشبه كثيرا بنية قطاع إقلاع FAT16، لكن كتلة FAT32 BPB تملك حقول إضافية. عناوين الإزاحة في قطاع الاقلاع أيضا تختلف عن نظيرتها في سجل إقلاع FAT16. علما أن أنظمة التشغيل التي لا تتوافق مع نظام ملفات FAT32 لا يمكنها قراءة وحدات تخزين FAT32.

بعد القطاعات المخفية 63 [32] تأتي قطاعات سجل الاقلاع (أي القطاعات المحجوزة) في وحدة التخزين وترتيبها سيكون بالشكل التالي:



08	07	06	05	04	03	02	01	00
17	16	15	14	13	12	11	10	09
26	25	24	23	22	21	20	19	18
35	34	33	32	31	30	29	28	27
43	42	42	41	40	39	38	37	36
53	52	51	50	49	48	47	46	45
62	61	60	59	58	57	56	55	54

سجل إقلاع FAT32	عنوان (خطي/ فيزيائي)	محتوى [35]		بايت	إزاحة
القطاع المنطقي 0 (قطاع الإقلاع)	CHS 0-1-1, LBA 63	تعليمة القفزة		3	000h (0)
		هوية صانعي القطع الأصلية OEM ID		8	003h (3)
		معاملات BPB		53	00Bh (11)
		معاملات الكتلة الممتدة EBPB		26	040h (64)
		شفرة إقلاع ابتدائية		420	05Ah (90)
		توقيع القطاع (علامة نهاية القطاع)		2	1FEh (510)
القطاع المنطقي 1 (قطاع معلومات نظام الملفات)	CHS 0-1-2, LBA 64	توقيع رئيس (توقيع أول)		4	000h (0)
		محجوزة		480	004h (4)
		توقيع بنية هذا القطاع (توقيع ثاني)		4	1E4h (484)
		حساب العناقيد الحرة		4	1E8h (488)
		رقم العنقود # (العنقود الحر التالي)		4	1ECh (492)
		محجوزة		12	1F0h (496)
		توقيع ذيلي (توقيع أخير)		4	1FCh (508)
القطاع المنطقي 2 أو 12 (قطاع شفرة الاقلاع)	CHS 0-1-3, LBA 65	شفرة إقلاع إضافية.	أو يتضمن فقط التوقيع إذا استخدم القطاع 12	510	000h (0)
		توقيع		2	1FEh (510)
القطاع المنطقي 6	CHS 0-1-7, LBA 69	قطاعات النسخة الاحتياطية			
القطاع المنطقي 7	CHS 0-1-8, LBA 70				
القطاع المنطقي 8	CHS 0-1-9, LBA 71				

قطاع إقلاع وحدة التحزين FAT32 (القطاع الأول أو القطاع 0 في سجل الاقلاع) [21]

وصف	مثال	طول / بايت	رمز تذكري	إزاحة
تعليمة القفزة (عادة تكون EB5890h وفق نظام الملفات وموقع شفرة الاقلاع). من أجل القفز إلى موقع آخر في البرنامج وتنفيذ التعليمات هناك. بمعنى آخر هذه التعليمة تأمر الحاسوب بتجاوز BPB و EBPB إلى أول بايت (عند الحيد 05Ah) في شفرة الاقلاع التنفيذية. ورغم أنها تتكون من 3 بايت لكن فقط 2 بايت الأولى هي التي تشكل التعليمة الفعلية (أي القفزة القصيرة) و EB 58 90 تفكك إلى JMP SHORT 58 NOP (قيمة 58 قد تكون مختلفة)، الخيارات الصالحة في بايت الأول هي EB أو E9 = (قفزة قريبة). قيمة / وظيفة / بايت EB — تعليمة القفزة — قفزة قصيرة SHORT JMP — 1 58 — إزاحة نسبية — 1 90 — تعليمة لا عملية NOP. — 1	EB 58 90	3	BS_jmpBoot	000h (0)
هوية صانعي القطع الأصلية (اسم ورقم إصدارة نظام التشغيل) OEM ID يشير إلى النظام المستخدم في تهيئة وحدة التخزين. عادة، يكون MSWIN4.1. لكن نظام التشغيل لا يستخدمها بعد التهيئة. بعض المصادر تعتبر هذا الحقل جزء من كتلة BPB	MSWIN4.1	8	BS_OEMName	003h (3)

بداية معاملات الكتلة BPB FAT32 (هذه الكتلة ستكون أكبر من الكتلة المعيارية). الكتلة BPB FAT32 هي نسخة ممتدة من كتلة BPB FAT16/FAT12. تتضمن معلومات متجانسة مع الكتلة المعيارية، لكنها أيضا تتضمن عدة حقول إضافية خاص بنظام FAT32. علما أن بعض إصدارات ويندوز OSR2 والإصدارات اللاحقة تستخدم بنية مختلفة من BPB، (سنذكرها هنا، فقط للإشارة).

المعاملات التالية أصلها من كتلة: DOS 3.31 BPB (FAT12, FAT16, FAT16B)

وصف	مثال	طول / بايت	رمز تذكري	إزاحة
عدد بايتات في كل قطاع. حجم القطاع على الوسيط الفيزيائي، في العادة يكون 512 بايت. ويمكن أن يكون أيضا بإحدى القيم التالية: عشري / ست عشري / ترتيب بايت 512 / 200h / 00 02 1024 / 400h / 00 04 2048 / 800h / 00 08 4096 / 1000h / 00 10 للتوافق يجب أن تكون دائما 512 بايت، غالبا الأنظمة لن تحاول فحص هذه القيمة لأن معظم شفرة FAT مصممة لاستخدام 512 بايت في كل قطاع. ورغم أن أنظمة مايكروسوفت تدعم القيم الأخرى، لكن لا ينصح باستخدامها.	00 02	2	BPB_BytsPerSec	00Bh (11)
عدد القطاعات المنطقية في كل عنقود. عدد القطاعات لكل وحدة تخصيص ؛ يجب أن يكون قوة العدد 2 أكبر من 0 ؛ عادة تكون بإحدى القيم التالية: عشري / ست عشري / بايت / كيلوبايت 1 / 1h / 512 / 0.5 2 / 2h / 1024 / 1 4 / 4h / 2048 / 2 8 / 8h / 4096 / 4 16 / 10h / 8192 / 8 32 / 20h / 16384 / 16 64 / 40h / 32768 / 32 128 / 80h / 65536 / 64 حجم العنقود المبدئي يعتمد على حجم وحدة التخزين. في ويندوز أكس بي تطبيق FAT32 يسمح بإنشاء وحدات تخزين بحجم أقصى 32 جيجابايت. على أية حال، وحدات التخزين الأكبر التي تنشأ في الأنظمة الأخرى (مثل ويندوز OSR2 95 واللاحقة) يمكن الوصول إليها أيضا في ويندوز أكس بي.	10	1	BPB_SecPerClus	00Dh (13)
عدد القطاعات المحجوزة. عدد القطاعات في المنطقة المحجوزة، بداية من القطاع المنطقي 0 ؛ يجب أن تكون 32 (رغم أنها قد تتفاوت حسب إعدادات الوسيط) على وحدات تخزين FAT32 (من 1-1-0 CHS إلى 32-1-0 CHS). هذه القطاعات قبل أول FAT1، منها ثلاثة قطاعات تستخدم من أجل سجل الاقلاع، وثلاثة أخرى تستخدم من أجل النسخة الاحتياطية من سجل الاقلاع، بقية القطاعات لا تستخدم وتبقى شاغرة. علما أن أول FAT1 تبدأ عند القطاع المنطقي 32 (أو عند القطاع المطلق 62 + 33 قطاع) = القطاع المطلق 95.	20 00	2	BPB_RsvdSecCnt	00Eh (14)
عدد نسخ FAT. عدد نسخ FATs على وحدة التخزين، عادة تكون 2. (تدعى مجازا : FAT1 و FAT2). عدة تطبيقات لن تعمل بالشكل الصحيح إذا القيمة لم تكون 2.	02	1	BPB_NumFATs	010h (16)
عدد مدخلات المجلد في الدليل الجذر (يجب أن تكون 0 على وحدات تخزين FAT32) هذا الحقل فقط في وسائط FAT12 و FAT16، ولا يستخدم في FAT32 أقراص FAT32 سوف تتجاهل هذا الحقل لأن الدليل الجذر على أقراص FAT32 لا يخزن في موقع ثابت، وسيكون في سلسلة عناقيد اعتيادية. ولا يوجد عدد أقصى لمدخلات (32-بايت) في الدليل الجذر، بالإضافة لذلك، حقل عنقود بداية الدليل الجذر يتضمن رقم أول عنقود في الدليل الجذر يسمح للدليل الجذر بالنمو عند الحاجة.	00 00	2	BPB_RootEntCnt	011h (17)

عدد القطاعات الإجمالي (16-بت) (القطاعات الصغرى < 32 ميغابايت) يجب أن تكون 0 على وحدات تخزين FAT32. هذا حقل عدد القطاعات الصغرى في FAT12/16، في وحدات تخزين FAT32. يستخدم حقل عدد القطاعات الإجمالي (32-بت). هذا الحقل فقط في وسائط FAT12 و FAT16، ولا يستخدم في FAT32	00 00	2	BPB_TotSec16	013h (19)
واصف الوسيط. بايت واصف الوسيط؛ عادة يكون F8h على جميع الوسائط الثابتة، و F0h على معظم الوسائط القابلة للفصل. وظيفة بايت واصف الوسيط تقديم معلومات عن الوسيط المستخدم؛ نوع القرص ونوع نظام الملفات. نسخة FAT1 تتضمن أيضا واصف وسيط في أول بايت من مدخلة العنقود 0. ويجب أن يماثل هذا الحقل في قطاع الاقلاع، نفس واصف الوسيط يجب أن يظهر كذلك في أول بايت في النسخة الثانية FAT2. في ويندوز تطبيقات ScanDisk تتفحص فقط واصف الوسيط في FAT1، وتتجاهل واصف الوسيط في قطاع الاقلاع، لهذا السبب لا يعتمد عليه في كشف الأخطاء. إذا كان هناك خطأ في قطاع الاقلاع. مدخلات واصف الوسيط كانت تستخدم في أقراص مايكروسوفت دوس FAT16، ولم تعد تستخدم في ويندوز	F8	1	BPB_Media	015h (21)
عدد القطاعات (16-بت) في كل نسخة FAT (يجب أن تكون 0 على وحدات تخزين FAT32). هذا حقل عدد قطاعات المشغولة من قبل كل FAT، لكن وحدات تخزين FAT32 تستخدم حقل عدد القطاعات لكل نسخة FAT 32-بت. هذا الحقل فقط في وسائط FAT12 و FAT16، ولا يستخدم في FAT32	00 00	2	BPB_FATSz16	016h (22)
عدد القطاعات في كل مسار. عدد القطاعات لكل مسار (لكل رأس) يتضمن قياسات القرص المستخدمة عند تهيئة القسم، في الأقراص التي تستخدم 13h ؛ عادة 63 (3Fh) على القرص الثابت. وحدة التخزين ستكون مجزأة إلى مسارات بمضاعفة الرؤوس والأسطوانات.	3F 00	2	BPB_SecPerTrk	018h (24)
عدد الرؤوس. عدد الرؤوس/الجوانب (لكل أسطوانة) يتضمن قياسات القرص المستخدمة عند تهيئة القسم، للأقراص التي تستخدم 13h؛ عادة تكون 255 (FFh) في معظم الأقراص الثابتة.	FF 00	2	BPB_NumHeads	01Ah (26)
عدد القطاعات المخفية (أسطوانة = 0 رأس = 0) على القرص. عدد القطاعات المخفية (1-0-0 CHS إلى 0-0-63) التي تسبق القسم الذي يتضمن وحدة التخزين ؛ عادة تكون 63 (3Fh) لأول وحدة تخزين. قيمة هذا الحقل تستخدم أثناء الاقلاع لحساب الحيد المطلق إلى الدليل الجذر ومناطق البيانات. عموما هذا الحقل ذو صلة فقط بالوسيط المرئي في INT 13h. ويجب أن يكون دائما 0 على الوسيط الذي بدون أقسام. بمعنى آخر : • تكون 63 قطاع في حالة القسم الأولي الأول في القرص الثابت أو القرص المنطقي الأول في القسم الممتد، • لكن في الأقسام الأولية: الثانية، الثالثة، الرابعة، أو القرص المنطقي الثاني أو أكثر في القسم الممتد، وحدات التخزين ستملك قيمة هنا تتضمن عدد قطاعات جميع الأقسام الأولية التي قبلها أو عدد القطاعات من بداية القسم الممتد. بالنسبة للقرص المنطقي. • القطاعات المخفية تدعى أيضا قسم غير دوس Non-DOS Partition (رغم أنه ليس قسم). وتدعى مسار 0 • بعض إصدارات ويندوز OSR2 واللاحقة تستخدم حقل إضافي يدعى حقل 2 بايت العليا من (قيمة) عدد القطاعات المخفية مع حقل عدد القطاعات المخفية. • عند مقارنة هذه المدخلة مع مدخلة سجل إقلاع FAT32 في ويندوز XP/2000. للقسم ممتد يتضمن نظام تشغيل (وليس فقط بيانات)، سوف تكتشف أن وحدة التخزين تحتفظ بالعدد الحقيقي للقطاعات المخفية قبلها (التي سوف لن تكون مخفية في هذه الحالة !) ؛ ليس فقط من بداية القسم الممتد. ولكن من أول قطاع على كامل القرص الفيزيائي ! وبالتالي، هناك إمكانية لإقلاع ويندوز 9x/ME من داخل القسم الممتد، إذا تم تعديل هذه المدخلة !	3F 00 00 00	4	BPB_HiddSec	01Ch (28)
عدد القطاعات الإجمالي في القسم (القطاعات الكبرى =>32 ميغابايت) لا يجب أن تكون صفر على وحدات التخزين FAT32. العدد الإجمالي للقطاعات (32-بت) على وحدات التخزين بحجم 65.536 قطاع أو أكثر. هذا الحقل يعرف أيضا باسم حقل " القطاعات الكبرى" أو "إجمالي القطاعات الكبرى" في وحدات تخزين FAT16. إذا كان الحقل 0، سوف يضمن التعداد في حقل عدد القطاعات الإجمالي (16-بت). • لنفترض أن لدينا قرص أو قسم بحجم قطاعات القرص: إذا كان حقل عدد القطاعات الإجمالي (أيا كان الحقل 16 بت أو 32 بت، بقيمة غير الصفر) أقل أو يساوي حجم القرص، فلا يعني ذلك وجود مشكلة مع وحدة التخزين FAT. في الواقع، ليس غريب أن تكون قيمة عدد القطاعات الإجمالي 16/32 بت أقل قليلا من حجم القرص. بل من الجيد أن تكون كذلك. هذا سيدل على أن هناك هدر في مساحة القرص. ولا يعني بالضرورة تلف وحدة تخزين FAT. لكن إذا كانت قيمة عدد القطاعات الإجمالي 16/32 بت أكبر من حجم القرص، فذلك يعني أن وحدة التخزين متضررة أو بتهيئة خاطئة لأنها في هذه الحالة تتجاوز نهاية الوسيط أو تتخطى البيانات التي تأتي بعدها على القرص. لذلك من الخطأ اعتبار وحدة التخزين صالحة إذا كان حجم الحقل (BPB_TotSec16/32) للوسيط أو القسم كبير جدا لأن ذلك يمكن أن يؤدي إلى ضياع في البيانات. • بعض إصدارات ويندوز OSR2 والإصدارات اللاحقة تستخدم حقل إضافي يدعى حقل 2 بايت العليا من (قيمة) عدد القطاعات الكبرى الإجمالي في القسم مع حقل عدد القطاعات الكبرى الإجمالي في القسم، بدلا من استخدام حقل عدد القطاعات الإجمالي 16-بت (الصغرى).	FC 8A 38 01	4	BPB_TotSec32	020h (32)

المعاملات التالية جزء من الكتلة الممتدة (غير كاملة) : DOS 7.1 EBPB (FAT32)

عدد القطاعات لكل FAT (القطاعات الكبرى 32-بت) عدد القطاعات المشغولة من كل نسخة FAT. فقط في وسائط FAT32. الحاسوب يستخدم هذا العدد مع عدد نسخ FATs وعدد القطاعات المحجوزة لتحديد بداية الدليل الجذر. الحاسوب يستطيع أيضا تحديد بداية	08 27 00 00	4	BPB_FATSz32	024h (36)

				منطقة بيانات المستخدم على وحدة التخزين وفقا لعدد المدخلات في الدليل الجذر. هذا الحقل لا يوجد في وسائط FAT12 و FAT16 ، ويعرف أيضا باسم حقل "القطاعات الكبرى في كل FAT". القيمة في حقل عدد القطاعات لكل نسخة جدول توزيع ملفات 16-بت يجب أن تكون 0 على وحدات تخزين FAT32. بعض إصدارات ويندوز OSR2 والإصدارات اللاحقة تستخدم حقل عدد القطاعات الكبرى لكل نسخة جدول توزيع ملفات 32-بت وحقل آخر يدعى حقل 2 بايت العليا من (قيمة) عدد القطاعات الكبرى في كل FAT. على قرص FAT32، بدلا من حقل عدد القطاعات لكل FAT.
028h (40)	BPB_ExtFlags	2	00 00	**أعلام ممتدة**. (من أجل تمكين/تعطيل مرآوية [15] نسخ FAT، علما أن وظيفة المرآوية تأتي في حالة تمكين مسبق)، يستخدم فقط في وحدة التخزين FAT32. وثائق مايكروسوفت بشأن FAT32 تقول أن هذا الحقل لا يستخدم في ويندوز أكس بي في المرآوية (أو النسخ الاحتياطي) [15] في قطاع الاقلاع: الثانية FAT2 تخزن مباشرة بعد الأولى FAT1، ويستخدمها نظام التشغيل في حالة تضرر النسخة الأولى ؛ في العادة هناك نسختين فقط يتم تحديثهما معا، لكن يمكن أن تكون هناك نسخ أكثر في القسم. • عند تمكين مرآوية FAT، كافة نسخ FAT (أي النسختين) يتم تحديثها (الكتابة في قطاع FAT1، تعني الكتابة إلى كل نسخة FAT أخرى. وأيضا يمكن قراءة قطاع FAT1 من أي نسخة مرآوية). بمعنى، تعدد FATs يجعلها نسخ احتياطية من بعضها البعض. • عند تعطيل مرآوية FAT، فقط النسخة النشيطة في حقل 3–0 ، يتم تحديثها. ويتم تجاهل الأخرى. • تعطيل المرآوية يسمح بمعالجة القرص الذي يملك قطاع فاسد في أحد نسخ FATs. إذا كان هناك قطاع فاسد، يمكن تعطيل الوصول إلى النسخة المتضررة بالكامل. ثم بناء FAT جديدة في إحدى نسخ FATs الغير نشيطة وجعلها قابلة للوصول بتغير قيمة النسخة النشيطة في هذا الحقل في قطاع الاقلاع. • ملحوظة :عادة هناك 8 بت في كل 1 بايت (وترقيمها من 0 إلى 7). البتات 3-0 : رقم يرتكز على الصفر (أي يبدأ من 0) من أجل النسخة النشيطة، ستكون صالحة فقط إذا تم تعيين بت 7 (أي تعطيل خاصية مرآوية (أي بت 7 = 1)) . إذا بت 7 كان خال (أي بت 7 = 0)، كافة نسخ FATs ستكون مرآوية كالمعتاد. البتات الأخرى محجوزة ويجب أن تكون 0. الجدول التالي يشرح ذلك بالتفصيل. أعلام معالجة مرآوية نسخ FAT : وصف | عدد البت (2 بايت) محجوزة دائما يبتدئ من 0 . تحفظ القيمة الحالية بعد ذلك. | 15–8 محجوزة. دائما تبدأ من 0 . تحفظ القيمة الحالية بعد ذلك | 6–4 رقم النسخة النشيطة FAT (إذا تم تعطيل المرآوية) | 3–0 بت 7 (م) | القيمة المرآوية زمن التشغيل على كافة نسخ FATs | 0 تمكين فقط نسخة FAT الوحيدة في بتات 3-0 هي النشيطة | 1 تعطيل 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0 0000h | النسخة النشيطة FAT | محجوزة | م | محجوزة إذا المرآوية معطلة بت 7 تمكين المرآوية (عند التعيين) أو تعطيلها (عند المسح)
02Ah (42)	BPB_FSVer	2	00 00	**رقم إصدارة نظام الملفات** (يستخدم فقط في وحدة التخزين FAT32. ويجب أن يكون 00h:00h). بايت عليا | بايت سفلى رقم مراجعة هامة (إصدارة هامة) | رقم مراجعة غير هامة (إصدارة ثانوية) كلتا إصدارات نظام الملفات (الهامة والثانوية) تعين إلى 0 في أنظمة ويندوز XP/ME/ 98/95 OSR2/2000 والأنظمة اللاحقة. هذه القيمة عادة لا تفحص، لكن، أحيانا محمل إقلاع مايكروسوفت يتفحصها، ويشتكي إذا لم تكن صفر، وكذلك إصدارات ويندوز القديمة سوف لن توصل وحدة التخزين إذا كانت بقيمة غير 0. هذا يدعم التوسع مستقبلا في نوع وسيط FAT32 دون القلق بشأن مشغلات FAT32 القديمة عند وصلها وحدة التخزين. • أدوات القرص ينبغي أن تتقيد بمضمون هذا الحقل ولا تعمل على وحدات التخزين ليست مصممة من أجلها، سواء كانت برقم إصدارة هامة أو ثانوية. وكذلك مشغلات FAT32 يجب أن تتفحص هذا الحقل ولا توصل وحدة التخزين إذا كانت لا تتضمن رقم إصدارة معروفة للمشغل (أي ضمن شفرة تصميم المشغل). • البايت الأكثر أهمية أو الأعلى، (high byte): هو بايت يتضمن 8 بت الأكثر أهمية MSB (في العادة أقصى اليسار) (من 8 إلى 15) مع اجتماع 2 بايت تمثل القيمة 16-بت (من 0 إلى 15 بت). • ملحوظة: OSR2 = Windows 95 OEM Service Release 2

02Ch (44)	BPB_RootClus	4	02 00 00 00	**عنقود بداية الدليل الجذر.** يستخدم فقط في وحدة التخزين FAT32. عادة يعين إلى 2 (أي العنقود 2، لكن ليس دائما). هذا هو رقم العنقود الخاص بالعنقود الأول من الدليل الجذر، (أي رقم العنقود الأول لبداية جدول الدليل الجذر). • سابقا، الدليل الجذر لم يكن جزء من منطقة البيانات، وكان بحجم وموقع ثابتين. لكن في نظام FAT32 الدليل الجذر يخزن كأي دليل آخر، في سلسلة العناقيد (في أي مكان في منطقة البيانات). يبدوا أن السبب في هذا توقع الزيادة في الحجم. قيمة هذا الحقل ينبغي أن تكون على الأقل 2. لكن ليس بالضرورة 2. • أدوات القرص التي تغير موقع الدليل الجذر يجب أن تضع العنقود الأول للدليل الجذر في أول عنقود سليم (أي، في العنقود 2، ما لم يكن فاسد) حتى تستطيع أدوات إصلاح القرص بسهولة إيجاد الدليل الجذر إذا أصبح هذا الحقل عرضيا بقيمة 0. • بعض إصدارات ويندوز OSR2 والإصدارات اللاحقة تستخدم حقل إضافي يدعى حقل 2 بايت العليا من حقل عنقود بداية الدليل الجذر مع حقل عنقود بداية الدليل الجذر.
030h (48)	BPB_FSInfo	2	00 00	**رقم قطاع معلومات نظام الملفات** (يستخدم فقط في وحدة التخزين FAT32). هذه الرقم يجب أن يشير إلى بنية القطاع الثاني الممتد الذي يدعى قطاع معلومات نظام الملفات، ضمن منطقة القطاعات المحجوزة في وحدة تخزين FAT32، عادة يكون 1. (الموقع وفق ترقيم القطاع النسبي الذي يبدأ من 0)؛ لأن تعداد القطاعات في نظام عنونة الكتل المنطقية LBA يبدأ من 0 ؛ لذلك العنوان 1 يساوي العنوان الفيزيائي CHS 0-1-2. ؛ ويحسب من بداية القطاع المنطقي 0. في القسم. إلى جانب القيمة 1. القيمة الممكنة الأخرى هي 0، وتعني أن قطاع معلومات نظام الملفات سيتم تجاهله (أي لا يستخدم). في هذه الحالة، نظام التشغيل سوف يعيد حساب المساحة الحرة على القرص كلما كانت هناك حاجة لهذه القيمة (أي قراءة كامل FAT)، عوض تخزين قيمة المساحة الحرة في قطاع معلومات نظام الملفات. (لكن هذا الأجراء سيكون أبطأ من عملية البحث عن القيمة) • بعض تطبيقات FAT32 تدعم جزئيا مواصفة مايكروسوفت لجعل قطاع معلومات نظام الملفات اختياريا بتحديد قيمة 0xFFFF (أو 0x0000) في هذه المدخلة (عند الحيد 0x030). • هذه مدخلة تشير إلى أحد القطاعات الإضافية في المنطقة المحجوزة على أقراص FAT32 يدعى أيضا قطاع BIGFATBOOTFSINFO ويتضمن قيم تستخدم في حساب العناقيد الحرة ورقم العنقود الخاص بالعنقود المخصص مؤخرا (أخر مرة)، هذه الحقول الإضافية تسمح للنظام استخدام القيم الموجودة بدون الحاجة إلى قراءة كامل FAT. • هناك نسخة من قطاع معلومات نظام الملفات في سجل الاقلاع الاحتياطي، لكن فقط النسخة التي يشير لها هذا الحقل ستبقى في حالة تحديث مستمر، ولا يتم تحديث أبدا بيانات قطاع معلومات نظام الملفات في النسخة الاحتياطية. • جميع بايتات الأخرى تعين إلى 0.. ويجب أن يكون قطاع معلومات النظام واحد على الأقل وبقع ضمن المنطقة المحجوزة. وأن لا تكون هذه القيمة مثل قيمة قطاع إقلاع النسخة الاحتياطية (عند الحيد 032h)، إذا لم تلبي هذه الشروط، لا يجب استخدام هذا القطاع، ما عدا ذلك، ينبغي أن يكون نظام الملفات ما زال صالح. إذا كان توقيع هذا القطاع غير صالح لا تستخدم معلومات هذا القطاع. علما أن مايكروسوفت قامت بتوثيق التوقيع الثاني فقط. أيضا، برامج تحليل القرص عادة تتفحص فقط رقم العناقيد الحرة وقيمة هذا الحقل تكون 1. • ملحوظة: (Aaron R. Reynolds الشفرة على اسم المبرمج هارون رينولدز = RRaA = AARD code) بنية قطاع معلومات نظام الملفات. في FAT32 (لمعلومات أكثر راجع جدول القطاع الثاني)
032h (50)	BPB_BkBootSec	2	06 00	**رقم قطاع النسخة الاحتياطية للسجل الاقلاع.** يستخدم فقط في وحدة التخزين FAT32. هذه النسخة تستخدم إذا تضررت النسخة الرئيسية من سجل الاقلاع. إذا لم تكن صفر، تشير إلى رقم قطاع النسخة الاحتياطية من سجل الاقلاع التي ستكون أيضا في منطقة المحجوزة في وحدة التخزين. عادة يكون القطاع 6. علما أن القيم الأخرى غير مستحسن هنا. وفق ترقيم القطاع النسبي الذي يبدأ من 0. (أي من القطاع المنطقي 0). القطاع 6 سيكون عند العنوان الفيزيائي CHS 0-1-7. • يمكنك مراقبة ومقارنة النسختين عند بدء التشغيل. إذا لم تتطابق، ستكون هناك رسالة تحذير. هذه النسخة قد لا تكون سليمة نتيجة تلف أو بسبب "فيروسات الاقلاع". • ولا ينبغي استخدام قطاع النسخة الاحتياطية، إذا تضمن هذا الحقل 0 أو رقم أكبر من أو يساوي عدد القطاعات المحجوزة أو تضمن نفس قيمة حقل قطاع معلومات نظام الملفات. • برنامج Fdisk عادة يصنع نسخة احتياطية من سجل الاقلاع في القطاع 6 في وحدة التخزين FAT32. وفي حالة استلم MBR خطأ في القراءة، أو واجه مشكلة في التحقق من صحة التوقيع عند قراءته قطاع الاقلاع، سوف يبحث في القطاع 6 ويحاول قراءة بقية محمل الاقلاع.
034h (52)	BPB_Reserved	12	00 00 00 00 00 00	**محجوزة** (اسم ملف إقلاع !). فقط في وحدة التخزين FAT32. يجب أن تكون دائما 0. برنامج تهيئة وحدات تخزين يعين هذا الحقل إلى 0

الإزاحة في القطاع	حجم	تعريف	
0x000	4	41615252h = "RRaA"	توقيع، ينبغي أن يكون:
0x004	480	محجوزة	
0x1E4	4	61417272h = "rrAa"	توقيع، ينبغي أن يكون:
0x1E8	4	عدد العناقيد الحرة (الشاغرة) على القرص، أو -1 إذا كانت مجهولة	
0x1EC	4	رقم العنقود المخصص حديثا	
0x1F0	12	محجوزة	
0x1FC	4	0x00 0x00 0x55 0xAA	توقيع (نهوي كبير)

	00 00 00 00 00 00			

نهاية معاملات الكتلة EBPB DOS 7.1 وبداية البقية المتممة لمعاملات الكتلة الممتدة FAT32 EBPB (وستكون بحجم 26 بايت) وفقا لوثائق مايكروسوفت. مايكروسوفت جعلت هذه كتلة متناقضة مع سابقتها بإضافتها 6 حقول جديدة بعد الحيد 20h، وتحريك 6 حقول أصلية (كانت من الحيد 24h إلى 36h ؛ منذ م.س-دوس 4) إلى نهاية EBPB (من 40h إلى 52h) :

حقل	إزاحة	بايت
رقم القرص الفيزيائي	40h	1
الرأس الحالي (أو محجوزة ؛ أعلام ويندوز أن تي)	41h	1
توقيع إقلاع ممتد	42h	1
الرقم التسلسلي لوحدة التخزين	43h	4
لصيقة اسم وحدة التخزين	47h	11
نوع نظام الملفات	52h	8

رقم القرص. رقم القرص المنطقي في دوس المرتبط برقم القرص الفيزيائي في نظام BIOS المستخدم في نداء المقاطعة INT 13h. ترقيم الأقراص المرنة يبدأ من 0x00 (دائما تأخذ "A" أو "B")، بينما الأقراص الثابتة تبدأ من 0x80. عادة، بغض النظر عن عدد الأقراص الثابتة الموجودة، لأن هذه القيمة لها علاقة فقط بقرص الاقلاع المحدد في نظام BIOS. يتم تعيين هذه القيمة قبل إصدار نداء المقاطعة INT 13 BIOS حتى يتم تحديد القرص الذي سيتم النفاذ إليه. نفس الحقل موجود في وسائط FAT12 و FAT16 ولكن في FAT32 عند حيد مختلف في قطاع الاقلاع	80	1	BS_DrvNum	040h (64)
محجوزة (سابقا: الرأس الحالي). (يجب أن تكون 0). (قد يكون هذا الحقل رقم البايت العليا للمدخلة السابقة !) نفس الحقل موجود في وسائط FAT12 و FAT16 ولكن في FAT32 عند حيد مختلف في قطاع الاقلاع هذه القيمة أصلا كانت ستستخدم لتخزين المسار الذي يقع عليه قطاع الاقلاع، لكن ويندوز أن تي يستخدمها لتخزين علمان من أعلام CHKDSK. • بت 0 (ترتيب منخفض): علم وحدة تخزين كثيرة الأخطاء dirty، يشير لضرورة تشغيل فحص آلي في الاقلاع التالي. • بت 1 (ترتيب منخفض) : علم أخطاء الإدخال/الأخراج I/O، ويشير لضرورة عمل تفحص للسطح في الاقلاع التالي. (البتات 7-2 ستكون دائما خالية). وينبغي لأدوات التهيئة تعين الحقل إلى 0. أصلا، كانت تستخدم لتخزين الأسطوانة/المسار الذي يقع عليه قطاع الاقلاع، لكن القيمة المخزنة على القرص حاليا لا تستخدم على هذا النحو.	00	1	BS_Reserved1	041h (65)
توقيع إقلاع ممتد نفس الحقل موجود في وسائط FAT12 و FAT16 ولكن في FAT32 عند حيد مختلف في قطاع الاقلاع يجب أن تكون 0x28 أو 0x29 حتى يتعرف عليه نظام ويندوز أن تي القيمة 29h تشير إلى وجود الحقول الثلاثة التالية. وإلا، ستكون 00h. حقل / حجم / إزاحة الرقم التسلسلي لوحدة التخزين / 4 بايت / 043h (67) لصيقة اسم وحدة التخزين / 11 بايت / 047h (71) نوع نظام الملفات / 8 بايت / 052h (82)	29	1	BS_BootSig	042h (66)
الرقم التسلسلي لوحدة التخزين. (رقم ست عشري عشوائي لكن فريد وبترتيب معكوس). يستخدم مع لصيقة اسم وحدة التخزين في تعقب وحدة التخزين على الوسيط القابل للفصل (أي للتمييز بين الأقراص). عادة، هذا الرقم يولد باستخدام التاريخ والوقت عند تهيئة وحدة التخزين، (باستخدام FORMAT.COM). نفس الحقل موجود في وسائط FAT12 و FAT16 ولكن في FAT32 عند حيد مختلف في قطاع الاقلاع	DF 0D 00 00	4	BS_VolID	043h (67)
لصيقة اسم وحدة التخزين. سلسلة نصية تستخدم محارف أسكي، وبايت الحشو 20h في الفراغات. هذا الحقل كان يستخدم سابقا لتخزين لصيقة وحدة التخزين، المخزنة الآن كملف خاص في الدليل الجذر؛ إذا لم تكن هناك لصيقة، يجب أن تكون : NO ␣ NAME ␣ ␣ ␣ ␣ نفس الحقل موجود في وسائط FAT12 و FAT16 ولكن في FAT32 عند حيد مختلف في قطاع الاقلاع	DRIVE C	11	BS_VolLab	047h (71)
نوع نظام الملفات. (هذا الحقل لغرض معلوماتي/إعلامي فقط). سلسلة محارف، محشوة بفراغات. تمثل هوية نظام الملفات. يجب أن تكون دائما : FAT32 ␣ ␣ ␣ نفس الحقل موجود في وسائط FAT12 و FAT16 ولكن في FAT32 عند حيد مختلف في قطاع الاقلاع كما يظهر من اسمها، في الواقع، هذه القيمة معلوماتية فقط وليست جزء من كتلة BPB ولا يستخدمها مشغل نظام ويندوز بعد تهيئة القرص، لأنها غالبا ستكون غير صحيحة، أو غير موجودة، بمعنى آخر، لا يمكن استخدمها في تحديد نوع نظام الملفات FAT12, FAT16, FAT32، لأن في مايكروسوفت نظام ملفات نوع FAT يحدد فقط عن طريق عدد العناقيد في القسم. رغم ذلك، يجب تعيين هذه السلسلة وفق نوع FAT لأن بعض مشغلات FAT من خارج مايكروسوفت ستتفحص الحقل	FAT32	8	BS_FilSysType	052h (82)

نهاية معاملات الكتلة الممتدة FAT32 EBPB وبداية شفرة الاقلاع وتوقيع الاقلاع

شفرة الاقلاع. شفرة برنامج الاقلاع، عادة برنامج IPL. ينقل التحكم إلى نظام التشغيل في القسم النشيط. على وحدة التخزين التي تقبل الإقلاع، المنطقة التي تتبع الكتلة الممتدة EBPB عادة تكون شفرة إقلاع تنفيذية. هذه الشفرة مسؤولة عن أداء كل ما هو ضروري كي تستمر عملية الاقلاع. في أنظمة ويندوز أن تي، شفرة الاقلاع هذه سوف تحدد موقع الملف NTLDR، وتحمله في الذاكرة، ثم تنقل عملية التنفيذ إلى الملف. بنية وعمل شفرة الاقلاع سيكون مختلف بين أنظمة التشغيل (دوس، ويندوز NT/95/98..الخ). في هذه المنطقة هناك دائما شفرة تنفيذية، حتى على القرص المرن الذي لا يقبل الإقلاع . هذه الشفرة ضرورية من أجل طباعة رسالة الخطأ المألوفة التي تظهر باللاتينية مثل "Non-system disk أو disk error" وتعني "ليس قرص نظام" أو "خطأ في القرص"، هذه ستكون في معظم الأقراص المرنة المعيارية المهيئة في م.س-دوس بدون استخدام الخيار "system".	متفاوت	420	BS_BootCode	05Ah (90)
توقيع الاقلاع. (يجب أن يكون الشفرة التنفيذية 55AAh. عندما يقرأ من تسجيل المعالج). في أي قطاع إقلاع 512 بايت دائما 2 بايت الأخيرة تحمل هذه القيمة (إذا كان القطاع صالح). حتى وإن كان القسم لا يستخدم كامل 512 بايت في القطاع. إذا اعتبرنا القطاع مضغوفة ثمانية، سيكون البايت 510 مساوي 0x55، والبايت 511 مساوي 0xAA. في وحدة تخزين FAT. (في القطاع 0). العديد من وثائق FAT تذكر أن التوقيع 0xAA55 يحتل 2 بايت الأخيرة من القطاع، لكن هذه المعلومة صحيحة فقط إذا كانت 512 بايت هو عدد بايتات لكل قطاع. إذا كان حجم القطاع أكبر من 512 بايت، لن تتغير إزاحة التوقيع (مع ذلك لا مشكلة أيضا إذا تضمن 2 بايت الأخيرة من قطاع الاقلاع هذا التوقيع)	55 AA	2	BS_Signature	1FEh (510)

وفق ألوان العمود الثاني في الجدول أعلاه:

بيانات هذه الحقول قد تتفاوت من حاسوب إلى آخر | بيانات هذه الحقول لا ينبغي أن تتفاوت بين أنظمة MSWIN4.1 | بيانات هذه قد تكون مختلفة في بعض الظروف، لكن عادة تبقى هي نفسها كما تظهر في الجدول

بنية قطاع الاقلاع 0 (في سجل إقلاع FAT32)

ما زالت جميع سجلات الاقلاع في أنظمة مايكروسوفت (منذ زمن دوس 2 والقرص المرن) تملك 3 بايت تدعى تعليمة القفزة، منها 2 بايت الأولى تشكل تعليمة القفزة الفعلية إلى بقية شفرة الاقلاع، والبايت الأخير تعليمة لا عملية NOP. مثال: EB 58 90 [3] بعد تعليمة القفزة تأتي هوية صانعي القطع الأصلية OEM ID أو اسم نظام التهيئة (بحجم 8 بايت) مثال:MSWIN4.1 ثم كتلة معاملات[20] BPB (التي لا تقبل التنفيذ).

كتلة BPB MSWIN 4.1 (كما تظهر أدناه) تملك حقول إضافية أخرى مقارنة بنظيرتها في قسم FAT16 (في ويندوز 95 سجل الاقلاع الأصلي MSWIN4.0). ورغم أن هذه الكتلة أطول من تلك التي في قسم FAT16.

ما زال آخر حقلين فيها هما : حقل لصيقة اسم وحدة التخزين " DRIVE C " وحقل هوية نظام الملفات " FAT32 ".

بنية قطاع إقلاع FAT32 (الجزء العلوي)

- تعليمة القفزة، اللا مشروطة (3 بايت)
- هوية صانعي القطع الأصلية (التعريف بنظام التهيئة) (8 بايت)

منطقة معاملات BPB

- عدد بايتات في كل قطاع (2 بايت)
- عدد القطاعات لكل عنقود (1 بايت)
- عدد القطاعات المحجوزة بداية من قطاع وحدة التخزين الأول (2 بايت)
- عدد نسخ FAT في نظام الملفات (1 بايت)
- عدد مدخلات الجذر المحتملة (2 بايت)
- العدد الإجمالي الأصغر للقطاعات 16-بت (القديم) (2 بايت)
- بايت واصف الوسيط (1 بايت)
- عدد قطاعات (16-بت) في كل نسخة FAT (2 بايت)
- عدد القطاعات لكل مسار وعدد القطاعات لكل رأس(2 بايت)
- عدد الرؤوس لكل أسطوانة (2 بايت)
- عدد القطاعات المخفية التي تسبق القسم (4 بايت)
- العدد الإجمالي الأكبر للقطاعات 32-بت (4 بايت)
- عدد قطاعات (32-بت) في كل نسخة FAT (4 بايت)
- أعلام مرآوية FAT (2 بايت)
- رقم إصدارة نظام الملفات (2 بايت)
- رقم العنقود الأول من الدليل الجذر (4 بايت)
- رقم قطاع معلومات نظام الملفات (2 بايت)
- رقم قطاع النسخة الاحتياطية لسجل الاقلاع (2 بايت)
- محجوزة (12 بايت)

في وحدة التخزين FAT32 الحقول التالية يجب تعيينها إلى الصفر: عدد مدخلات الدليل الجذر (في FAT12/FAT16)، والعدد الإجمالي الأصغر للقطاعات 16-بت، وعدد القطاعات في كل FAT (فقط في FAT12/FAT16).

منطقة معاملات EBPB

- رقم القرص الفيزيائي (1 بايت)
- محجوزة (1 بايت)
- توقيع إقلاع ممتد (1 بايت)
- الرقم التسلسلي لوحدة التخزين (4 بايت)
- لصيقة اسم وحدة التخزين (في القسم) (11 بايت)
- نوع نظام الملفات (8 بايت)

منطقة الشفرة

- شفرة إقلاع تنفيذية. سوف تكون مختلفة بين أنظمة التشغيل،

CHS 0-1-**1**, LBA **63** (القطاع المطلق)

```
      00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F   0123456789ABCDEF
0000: EB 58 90 4D 53 57 49 4E 34 2E 31 00 02 10 20 00 [ ëX MSWIN4.1      ]
0010: 02 00 00 00 00 F8 00 00 3F 00 FF 00 3F 00 00 00 [      ø  ? ÿ ?    ]
0020: FC 8A 38 01 08 27 00 00 00 00 00 00 02 00 00 00 [ üŠ8  '           ]
0030: 00 00 06 00 00 00 00 00 00 00 00 00 00 00 00 00 [                  ]
0040: 80 00 29 DF 0D 00 00 44 52 49 56 45 20 43 20 20 [ € )ß   DRIVE C   ]
0050: 20 20 46 41 54 33 32 20 20 20 FA 33 C9 8E D1 BC [   FAT32   ú3ÉŽÑ¼ ]
0060: F8 7B 8E C1 BD 78 00 C5 76 00 1E 56 16 55 BF 22 [ ø{ŽÁ½x Åv  V U¿" ]
0070: 05 89 7E 00 89 4E 02 B1 0B FC F3 A4 8E D9 BD 00 [  ‰~ ‰N ± üó¤ŽÙ½  ]
0080: 7C C6 45 FE 0F 8B 46 18 88 45 F9 38 4E 40 7D 25 [ |ÆEþ ‹F ^Eù8N@}% ]
0090: 8B C1 99 BB 00 07 E8 97 00 72 1A 83 EB 3A 66 A1 [ ‹Á™»  è- r fë:f¡ ]
00A0: 1C 7C 66 3B 07 8A 57 FC 75 06 80 CA 02 88 56 02 [  |f; ŠWüu €Ê ^V  ]
00B0: 80 C3 10 73 ED BF 02 00 83 7E 16 00 75 45 8B 46 [ €Ã sí¿  ƒ~  uE‹F ]
00C0: 1C 8B 56 1E B9 03 00 49 40 75 01 42 BB 00 7E E8 [  ‹V ¹  I@u B» ~è ]
00D0: 5F 00 73 26 B0 F8 4F 74 1D 8B 46 32 33 D2 B9 03 [ _ s&°øOt ‹F23Ò¹  ]
00E0: 00 3B C8 77 1E 8B 76 0E 3B CE 73 17 2B F1 03 46 [  ;Èw ‹v ;Îs +ñ F ]
00F0: 1C 13 56 1E EB D1 73 0B EB 27 83 7E 2A 00 77 03 [  V ëÑs ë'ƒ~* w  ]
0100: E9 FD 02 BE 7E 7D AC 98 03 F0 AC 84 C0 74 17 3C [ éý ¾~}¬˜ ð¬„Àt  ]
0110: FF 74 09 B4 0E BB 07 00 CD 10 EB EE BE 81 7D EB [ ÿt ´ »  Í ëî¾ }ë ]
0120: E5 BE 7F 7D EB E0 98 CD 16 5E 1F 66 8F 04 CD 19 [ å¾ }ëà˜Í ^ f  Í  ]
0130: 41 56 66 6A 00 52 50 06 53 6A 01 6A 10 8B F4 60 [ AVfj RP Sj j ‹ô` ]
0140: 80 7E 02 0E 75 04 B4 42 EB 1D 91 92 33 D2 F7 76 [ €~  u ´Bë ‘’3Ò÷v ]
0150: 18 91 F7 76 18 42 87 CA F7 76 1A 8A F2 8A E8 C0 [  '÷v B‡Ê÷v ŠòŠèÀ ]
0160: CC 02 0A CC B8 01 02 8A 56 40 CD 13 61 8D 64 10 [ Ì  Ì¸   ŠV@Í a d  ]
0170: 5E 72 0A 40 75 01 42 03 5E 0B 49 75 B4 C3 03 18 [ ^r @u B ^ Iu´Ã    ]
0180: 01 27 0D 0A 4E 6F 20 4F 53 20 46 6F 75 6E 64 20 [  ' No OS Found  ]
0190: 20 20 20 20 20 20 20 FF 0D 0A 44 69 73 6B 20 49 [       ÿ Disk I ]
01A0: 2F 4F 20 45 72 72 6F 72 FF 0D 0A 49 6E 73 65 72 [ /O Errorÿ Inser ]
01B0: 74 20 4F 53 20 73 65 74 75 70 20 64 69 73 6B 2C [ t OS setup disk, ]
01C0: 20 74 68 65 6E 20 70 72 65 73 73 20 61 6E 79 20 [  then press any ]
01D0: 6B 65 79 0D 0A 00 00 00 49 4F 20 20 20 20 20 20 [ key     IO      ]
01E0: 53 59 53 4D 53 44 4F 53 20 20 20 53 59 53 7E 01 [ SYSMSDOS   SYS~  ]
01F0: 00 57 49 4E 42 4F 4F 54 20 53 59 53 00 00 55 AA [  WINBOOT SYS  Uª ]
      00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F
```

قطاع إقلاع وحدة التخزين FAT32، وبندوز 98/98SE/ME، كل سطر 16 بايت (32 سطر). عند 63 LBA (CHS 0-1-1)

بنية قطاع الاقلاع FAT32 (الجزء السفلي)

- رسائل الأخطاء وأسماء ملفات النظام الثلاثة (الأسماء قد تتفاوت وفق نظام التشغيل المستخدم في FAT32.)
- مواقع البيانات (إزاحة الرسالة) (4 بايت)
- أسماء ملفات النظام الثلاثة (11 بايت)
- رسائل الأخطاء (84 بايت !)
- مواقع البيانات (إزاحة الملف) (3 بايت)
- أسماء ملفات النظام الثلاثة (11 بايت)
- أسماء ملفات النظام الثلاثة (11 بايت)
- توقيع قطاع الاقلاع ؛ سيكون دائما 55AAh (إذا كان القطاع صالح !).

130 بايت الأخيرة من أول قطاع في سجل الاقلاع تتضمن بعض مواقع البيانات (03 18 01 27)، رسائل الأخطاء، وأسماء ملفات النظام الثلاثة (بداية من ملف IO.SYS) وتوقيع الاقلاع الأخير (الرقم السحري) AA55h؛

تذكر أن قيم الست عشري (في البيانات العددية التي تتطلب أكثر من بايت واحد) في أنظمة x86 دائما تخزن في الذاكرة بايت الأدنى أولا وبايت الأعلى أخيرا.

قطاع معلومات نظام الملفات (القطاع الثاني في سجل إقلاع FAT32)

هذا القطاع الثاني [16][17] (أي القطاع 1) من 3 قطاعات تشكل طول سجل إقلاع القسم FAT32. هذا القطاع يتضمن بعض بيانات المتقلبة باستمرار في FAT32. ووضع في الخدمة لتحسين أداء بعض العمليات (مثل, الحصول على قيمة المساحة الحرة). قطاع معلومات نظام الملفات ليس جزء من كتلة FAT32 EBPB، لكنه يملك فيها مؤشر عند الحيد 0x30. في قطاع الاقلاع (أي القطاع 0).

كلتا النسختين الأولية والاحتياطية من سجل الاقلاع تشير إلى نفس قطاع معلومات نظام الملفات. لكن فقط النسخة التي يشير لها هذا الحقل سيتم تحديثها باستمرار، وهذا يفسر سبب الاختلاف بين النسختين الأولية والاحتياطية كما ذكرنا سابقا عن حقل 4 بايت في قطاع معلومات نظام الملفات.

بنية قطاع معلومات نظام الملفات FAT32 في القطاع الثاني (القطاع 1) عند LBA 64 (CHS 0-1-2)

وصف	مثال	طول / بايت	رمز تذكري	إزاحة
توقيع رئيس (توقيع أول). تعيين هوية قطاع معلومات نظام الملفات. توقيع قطاع معلومات نظام الملفات FSI. النظام يستخدم هذا الوسم (التوقيع) كي يتحقق من صحة هوية هذا القطاع. الذي يجب أن يكون 52526141h. في MSWIN4.1، الوسم يعرف القطاع باسم: "سجل إقلاع ممتد" طالما قطاع معلومات نظام الملفات يقع في القطاع المنطقي 1، الذي عادة هو الموقع الذي تبدأ فيه نسخة FAT في أنظمة FAT12 و FAT16 (التي تستخدم قطاع واحد فقط)، وجود هذا التوقيع يمنع نسخ دوس القديمة من وصل وحدة تخزين FAT32، لأنها تتوقع من القيم في العنقود 0 و العنقود 1 أن تتبع متسلسلات ثنائية معينة والتي لن تكون مع هذا التوقيع.	ست عشري: 52526141h / شفرة أسكي: RRaA	4	FSI_LeadSig	000h (0)
محجوزة. يجب أن تكون 0. عند التهيئة، ولا تتغير بعد ذلك أبدا (480 بايت جميعها أصفار NULL) ينبغي دائما على برامج تهيئة FAT32 تعين جميع بايتات هذا الحقل إلى 0. ولا تستخدم أبدا بعد ذلك.	00 00 00 00 00 00 00 00 00 00 00 00	480	FSI_Reserved1	004h (4)
توقيع بنية هذا القطاع (توقيع ثاني). تعيين هوية بداية البيانات توقيع قطاع معلومات نظام الملفات FSI يتحقق من صحة القطاع. ويجب أن يكون 72724161h. هذا وسم آخر يحدد بداية بيانات إجمالي العناقيد الحرة والعنقود التالي المتوفر (هذا توقيع محلي يحدد موقع الحقول المستخدمة).	ست عشري: 72724161h / شفرة أسكي: rrAa	4	FSI_StrucSig	1E4h (484)
حساب العناقيد الحرة. هذا الحقل يتضمن الحساب الأخير المعروف لعدد عناقيد البيانات الحرة (الشاغرة) على وحدة التخزين. إذا كانت القيمة 0xFFFFFFFF أو -1 يعني ذلك أن عدد العناقيد الحرة مجهول، ويجب حسابها. القيمة لن تكون صحيحة بالضرورة. وعلى نظام التشغيل التأكد من صحتها قبل استخدمها وتأكد أنها على الأقل أصغر أو تساوي حساب العناقيد في وحدة التخزين. أثناء التهيئة يجب تعينها إلى 0xFFFFFFFF وتحديثها فيما بعد من قبل النظام.	EF 87 04 00	4	FSI_Free_Count	1E8h (488)
رقم العنقود # (العنقود الحر التالي). الرقم إشارة إلى المكان حيث يجب على مشغل النظام بدأ البحث فيه عن العنقود الحر. عادة يعين الحقل إلى رقم عنقود البيانات الأخير، المخصص من قبل المشغل. إذا كانت القيمة FFFFFFFFh، لن تكون هناك إشارة (أي الحقل غير معين) وعلى النظام البدء بالبحث عند العنقود 2. (أي 0x00000002). أثناء التهيئة يجب تعينها إلى FFFFFFFFh وتحديثها فيما بعد من قبل نظام التشغيل. يمكن استخدام أي قيمة أخرى، ولكن على نظام التشغيل تفحصها أولا قبل استخدمها وتأكد أنها رقم عنقود صالح على وحدة التخزين. نظرا لكبر جدول FAT في وحدة تخزين FAT32، يستغرق وقتا طويلا إذا كان عدد العناقيد المخصصة كبير عند بداية FAT والمشغل يبدأ بالبحث عن العنقود الحر من بداية العنقود 2.	05 1C 09 00	4	FSI_Nxt_Free	1ECh (492)
محجوزة. دائما 0 عند التهيئة، لكن لا يعتمد عليها ولا تتغير بعد ذلك أبدا (12 بايت جميعها أصفار NULL) ينبغي دائما على برامج تهيئة FAT32 تعيين جميع بايتات هذا الحقل إلى 0 ولا تستخدم أبدا بعد ذلك.	00 00 00 00 00 00 00 00 00 00 00 00	12	FSI_Reserved2	1F0h (496)
توقيع ذيلي (توقيع أخير). توقيع قطاع معلومات نظام الملفات لتحقق أن هذا القطاع هو قطاع معلومات نظام الملفات، ويجب أن تكون القيمة 4 بايت (00 00 55 AA)، حتى يعتبر مضمون القطاع صالح. 2 بايت العليا في هذه القيمة تحتل نفس موقع إزاحة البايت 510 و 511 من عند 1FEh المستخدمة في توقيع القطاع 0	00 00 55 AA	4	FSI_TrailSig	1FCh (508)

بنية قطاع معلومات نظام الملفات (القطاع الثاني في سجل إقلاع FAT32)

- توقيع قطاع معلومات نظام الملفات (توقيع رئيسي) (4 بايت)
- عدد العناقيد الحرة (4 بايت)
- محجوزة (480 بايت)
- رقم العنقود # (4 بايت)
- توقيع قطاع معلومات نظام الملفات (توقيع البنية) (4 بايت)
- محجوزة (12 بايت)
- توقيع قطاع معلومات نظام الملفات (توقيع ذيلي) (4 بايت)

(القطاع المطلق) CHS 0-1-2, LBA **64**

```
          00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F  0123456789ABCDEF
0000: 52 52 61 41 00 00 00 00 00 00 00 00 00 00 00 00 [RRaA............]
0010: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [................]
*  [34]
01E0: 00 00 00 00 72 72 61 41 EF 87 04 00 05 1C 09 00 [....rrAa....T...]
01F0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 55 AA [..............U.]
```

قطاع معلومات نظام الملفات وحدة التخزين FAT32، وبندوز 98/98SE/ME.

معلومات القطاع قد تكون قديمة ولا تعكس مضمون الوسيط الحالي، هذا لأن أنظمة التشغيل ليست جميعها تحدث أو تستخدم هذا القطاع، وحتى وإن فعلت ذلك، لن يكون المحتوى صالح إذا تم إزالة الوسط بدون فصل وحدة التخزين بشكل صحيح أو بعد انقطاع التيار الكهربائي. لهذا يجب على أنظمة التشغيل التحقق أولا من الأعلام الثنائية الاختبارية الخاصة بوضعية إيقاف التشغيل في وحدة التخزين، المتواجدة في مدخلة FAT في العنقود 1 أو FAT32 EBPB عند الحيد 0x041 وتجاهل البيانات المخزنة في قطاع معلومات نظام الملفات، إذا كانت تلك الأعلام الثنائية تشير إلى فصل وحدة التخزين سابقا بشكل غير صحيح. في دوس و وبندوز برنامج ScanDisk يتحقق من حجم المساحة الحرة في القسم (عدد العناقيد الحرة)، ويصحح القيمة (إذا كانت خاطئة) المخزنة في قطاع معلومات نظام الملفات ، لكن لا يفعل ذلك إذا كان المؤشر هو 0 في قطاع الاقلاع؛ وهذا يمنع الأخطاء.

مثل على حساب إجمالي العناقيد الحرة والعنقود التالي المتوفر

نفترض أن هناك قرص يملك 4 كيلوبايت (4096 بايت) لكل عنقود، حجم المساحة الحرة على القرص سيكون بحساب البايت التالي:

وصف		عشري	ست عشري
ثبت أن رقم العنقود 596,997 هو العنقود التالي المتوفر على القرص. ويبدوا أن ويندوز يقوم بتحديث هذان الموقعين في كل مرة يكتب فيها إلى القرص!	عدد العناقيد الحرة (الحساب الأخير المعروف)	296,943	000487EF
296,943 عنقود * 4096 بايت (أو للعنقود) = 1,216,278,528 بايت (حوالي 1.13 جيجابايت)	العنقود التالي المتوفر (إشارة إلى مكان البحث)	596,997	00091C05
بعد مرور تقريبا شهر، تم فحص المواقع مرة أخرى (من 1E8 إلى 1EF) ووجد أنها تتضمن هذه المرة: DF EB 03 00 54 0A 0B 00 وتعني المساحة الحرة:	عدد العناقيد الحرة (الحساب الأخير المعروف)	256,991	0003EBDF
256,991 قطاع * 4096 بايت (أو للعنقود) = 1,052,635,136 بايت (حوالي 0.97 جيجابايت)؛ العنقود التالي المتوفر كان في 723,540.	العنقود التالي المتوفر (إشارة إلى مكان البحث)	723,540	000B0A54

أخيرا، القطاع الثالث والأخير في سجل الاقلاع، يتضمن بقية الشفرة التنفيذية مع توقيع AA55 h الذي يظهر دائما في نهاية كل قطاع في سجل الاقلاع.

قطاع شفرة إقلاع FAT32 (القطاع الثالث في سجل إقلاع FAT32)

القطاع الثالث من القسم (CHS 0-1-3)، يتضمن شفرة الاقلاع (البرنامج، تقريبا 510 بايت)، تبدأ بالوسم 0F B6 66 (رغم أنه ليس فريد) وتنتهي بتوقيع إقلاع 55AA. هذا القطاع لا يتضمن معلومات تخص النظام، يتضمن فقط شفرة معيارية متجانسة على كافة أقراص FAT32. إذا تم إنشاء أقسام FAT32 في ويندوز أكس بي أو ويندوز 7، القطاع الثالث في سجل الاقلاع الجديد لن يتضمن أية شفرة؛ وسيكون محشو ببايت 00، باستثناء التوقيع الأخير 55h AAh، لأن مايكروسوفت تستخدم القطاع 12 بدلا من القطاع الثالث في منطقة المحجوزة من أجل محمل إقلاع متمدد.

بنية قطاع شفرة إقلاع FAT32، في القطاع 2. (القطاع الأخير من 3 قطاعات تشكل سجل إقلاع قسم FAT32)

وصف	مثال	بايت	رمز تذكري	إزاحة
شفرة برنامج الاقلاع (الأمثلة أدناه)	متفاوت	510	BS_BootCode	000h
التوقيع ؛ يجب أن يكون 55 AAh	55 AA	2	BS_Signature	1FEh (510)

CHS 0-1-3, LBA **65** (القطاع المطلق)

```
   00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F  0123456789ABCDEF
0000: FA 66 0F B6 46 10 66 8B 4E 24 66 F7 E1 66 03 46 [úf.¶F.f‹N$f÷áf.F]
0010: 1C 66 0F B7 56 0E 66 03 C2 33 C9 66 89 46 FC 66 [.f.·V.f.Â3Éf‰Füf]
0020: C7 46 F8 FF FF FF FF FA 66 8B 46 2C 66 83 F8 02 [ÇFø....úf‹F,fƒø.]
0030: 0F 82 CF FC 66 3D F8 FF FF 0F 0F 83 C5 FC 66 0F [.‚Ïüf=ø....ƒÅüf.]
0040: A4 C2 10 FB 52 50 FA 66 C1 E0 10 66 0F AC D0 10 [¤Â.ûRPúfÁà.f.¬Ð.]
0050: 66 83 E8 02 66 0F B6 5E 0D 8B F3 66 F7 E3 66 03 [fƒè.f.¶^.‹óf÷ãf.]
0060: 46 FC 66 0F A4 C2 10 FB BB 00 07 8B FB B9 01 00 [Füf.¤Â.û»..‹û¹..]
0070: E8 BE FC 0F 82 AA FC 38 2D 74 1E B1 0B 56 BE D8 [è¾ü.‚ªü8-t.±.V¾Ø]
0080: 7D F3 A6 5E 74 19 03 F9 83 C7 15 3B FB 72 E8 4E [}ó¦^t..ùƒÇ.;ûrèN]
0090: 75 D6 58 5A E8 66 00 72 AB 83 C4 04 E9 64 FC 83 [uÖXZèf.r«ƒÄ.édüƒ]
00A0: C4 04 8B 75 09 8B 7D 0F 8B C6 FA 66 C1 E0 10 8B [Ä.‹u.‹}.‹ÆúfÁà.‹]
00B0: C7 66 83 F8 02 72 3B 66 3D F8 FF FF 0F 73 33 66 [Çfƒø.r;f=ø...s3f]
00C0: 48 66 48 66 0F B6 4E 0D 66 F7 E1 66 03 46 FC 66 [HfHf.¶N.f÷áf.Füf]
00D0: 0F A4 C2 10 FB BB 00 07 53 B9 04 00 E8 52 FC 5B [.¤Â.û»..S¹..èRü[]
00E0: 0F 82 3D FC 81 3F 4D 5A 75 08 81 BF 00 02 42 4A [.‚=ü ?MZu. ¿..BJ]
00F0: 74 06 BE 80 7D E9 0E FC EA 00 02 70 00 03 C0 13 [t.¾€}é.üê..p..À.]
0100: D2 03 C0 13 D2 E8 18 00 FA 26 66 8B 01 66 25 FF [Ò.À.Òè..ú&f‹.f%.]
0110: FF FF 0F 66 0F A4 C2 10 66 3D F8 FF FF 0F FB C3 [...f.¤Â.f=ø...ûÃ]
0120: BF 00 7E FA 66 C1 E0 10 66 0F AC D0 10 66 0F B7 [¿.~úfÁà.f.¬Ð.f.·]
0130: 4E 0B 66 33 D2 66 F7 F1 66 3B 46 F8 74 44 66 89 [N.f3Òf÷ñf;Føtdf‰]
0140: 46 F8 66 03 46 1C 66 0F B7 4E 0E 66 03 C1 66 0F [Føf.F.f.·N.f.Áf.]
0150: B7 5E 28 83 E3 0F 74 16 3A 5E 10 0F 83 A4 FB 52 [·^(ƒã.t.:^..ƒ¤ûR]
0160: 66 8B C8 66 8B 46 24 66 F7 E3 66 03 C1 5A 52 66 [f‹Èf‹F$f÷ãf.ÁZRf]
0170: 0F A4 C2 10 FB 8B DF B9 01 00 E8 B4 FB 5A 0F 82 [.¤Â.û‹ß¹..è´ûZ.‚]
0180: 9F FB FB 8B DA C3 00 00 00 00 00 00 00 00 00 00 [ŸûûÚÃ..........]
0190: 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 [..............Uª]
```

قطاع شفرة إقلاع، وحدة التخزين FAT32، ويندوز ME/98SE/98،

القطاع الثالث لا يتضمن معلومات تخص النظام، يعني أن المحتويات دائما نفسها في أي نظام من نفس النوع. أي إذا تعرض هذا القطاع إلى التلف (مع النسخة الاحتياطية عند LBA 8)، يمكنك الحصول على نسخة من قطاع أي وحدة تخزين FAT32 أخرى واستخدامها لاستعادة القطاع المتضرر.

بقية القطاعات المحجوزة لسجل إقلاع FAT32 ستكون محشوة بأصفار إلى أخر حيد 3FF0 في القطاع (16368 بايت) في القطاع 32 (القطاع المنطقي 31).

النسخة الاحتياطية من سجل إقلاع FAT32

من إحدى مميزات وحدات تخزين FAT32، وجود حقل لرقم قطاع النسخة الاحتياطية من سجل الاقلاع في المنطقة المحجوزة. هذه الميزة غير موجودة في وحدات تخزين FAT12 و FAT16 لذلك يمكن خسارتها بالكامل إذا تم الكتابة على القطاع 0 أو في حالة تلف القطاع، ووجود حقل رقم قطاع النسخة الاحتياطية. يقلل من خطورة هذا المشكلة في وحدات تخزين FAT32.

في حالة الكتابة فوق القطاع 0، يمكن لأداة إصلاح القرص استعادة قطاع الاقلاع من النسخة الاحتياطية. أما في حالة تلف القطاع 0، يمكن وصل وحدة التخزين كي يستطيع المستخدم الوصول إلى البيانات قبل استبدال القرص. الحالة الثانية (أي حالة تلف القطاع 0) هي سبب وجود القيمة 6 وليس قيمة أخرى في حقل رقم قطاع النسخة الاحتياطية. لأنه إذا كان القطاع 0 غير صالح للقراءة، عدة أنظمة تشغيل [22] مبرمجة Hard coded كي تتفحص قطاع الاقلاع الاحتياطية عند القطاع 6 على وحدة تخزين FAT32. علما أن من بداية قطاع النسخة الاحتياطية يعتبر سجل إقلاع كامل.

في مايكروسوفت قطاع الاقلاع FAT32 بطول 3 قطاعات (512 * 3). والنسخة الاحتياطية من 3 قطاعات تبدأ عند قطاع النسخة الاحتياطية. وتتضمن أيضا نسخة احتياطية من قطاع معلومات نظام الملفات، حتى وإن كان حقل قطاع معلومات نظام الملفات في هذه النسخة الاحتياطية يشير إلى نفس القيمة كما هي مخزنة في كتلة BPB في القطاع 0.

عمل الشفرة (نظام ملفات FAT32)

في ويندوز 95 :

سجل الاقلاع الرئيسي يحمل القطاع الأول من سجل الاقلاع في موقع الذاكرة المعتاد 0000:7C00. عناوين الذاكرة من 7C0Bh إلى 7C59h، سوف تحتلها الكتلة BPB و 292 بايت التالية (العناوين من 7C5A إلى 7D7D) تتضمن الشفرة التنفيذية لهذا القطاع (أي البرنامج الرئيسي وعدة روتينات ثانوية) التي بدورها سوف تحمل شفرة الاقلاع من القطاع الثالث، الأخير (في سجل إقلاع وحدة التخزين FAT32)، والذي يتضمن معظم الشفرة المستخدمة في قراءة مدخلات 32-بت على القرص الثابت، كي يستطيع نظام دوس أو ويندوز تحميل أجزء الملف IO.SYS في الذاكرة (التي أخيرا ستحمل بقية نظام التشغيل).

في ويندوز 98 :

بعد أن يحمل سجل الاقلاع الرئيسي قطاع الاقلاع في موقع الذاكرة 000:7C00، يستخدم تعليمة العودة RET لنقل التنفيذ إلى شفرة قطاع الاقلاع، تقريبا الروتين سيكون كالتالي :

1. حفظ عنوان جدول معاملات القرص DPT الذي يشير له متجه المقاطعة INT 1Eh.
2. تغيير متجه المقاطعة INT 1Eh كي يشير إلى الجدول المعدل DPT عند 0000:0522.
3. نسخ جدول DPT إلى عنوان 0000:0522.
4. تغيير نسخة DPT.
5. إذا كان القرص قرص ثابت، أقرأ سجل MBR وأحضر مؤشر النظام للقسم الحالي.
6. حمل القطاعين التاليين في موقع الذاكرة 0000:7E00 (تنتهي عند 0000:81FF).
   - إذا حدث خطأ في القراءة، جرب النسخة الاحتياطية من القطاعات.
   - إذا فشلت المحاولة، أعرض رسالة الخطأ "Disk I/O error" متبوعة برسالة "Replace the disk, and then press any key" وبعد ضغط المفتاح، النظام يعيد التشغيل.
1. استمر في تنفيذ عند موقع الذاكرة 0000:8000 (لفهم بقية العملية... أبحث في الأنترنت باللغة الانجليزية. عن تفكيك شفرة قطاع إقلاع ويندوز 98 الإصدارة الثانية)

البرنامج سوف يتحقق أيضا من العناصر التالية:

- عدد القطاعات لكل FAT يجب أن يكون صفر (فقط في FAT12/FAT16).
- يجب أن تكون هناك نسخة احتياطية من قطاع الاقلاع.
- عدد القطاعات في المنطقة المحجوزة يجب أن يكون أكبر من 3.
- رقم إصدارة نظام الملفات يجب أن يكون 0 (فقط في FAT32)

إذا كان أحد المعطيات خاطئ يعرض رسالة "No OS Found" أو "Invalid system disk" متبوعة برسالة "Replace the disk, and then press any key"بعد ضغط المفتاح النظام يعيد التشغيل.

قطاع الاقلاع في أنظمة دوس الأخرى

جهاز أتاري أس تي

أتاري أس تي Atari ST كان حاسوب منزلي من أتاري Atari، من عام 1985. مع معالج موتورولا 68000، المحارف "ST" اختصار 16/32، وتشير إلى معالج 32-بت وناقل بيانات خارجي 16-بت. الطراز الأول للحاسوب 520ST كان يملك قرص مرن خارجي ووحدة طاقة، أصبحت فيما بعد مدمجة في جهاز 1040ST. ثم ذاكرة RAM بحجم 512 كيلوبايت في جهاز 520 و 520STFM، ثم أصبحت بحجم 1 ميغابايت في جهاز 1040. ثم حدثت عدة ترقيات، حتى عام 1993 ترتكز على جهاز أتاري فالكون مع معالج موتورولا 68030. بعد إصدار حاسوب أتاري أس تي، الشركة أنتجت أيضا أتاري أس تي إي، و أتاري تي تي، وأتاري مجا أس تي إي وأتاري فالكون 030، (الذي كان أخر منتج من أتاري؛ وإستخدم نظام مالتي توس الذي يسمح بتعدد المهمات الوقائي).

بسبب واجهة المستخدم، التي تشبه الماكنتوش Macintosh البعض من باب السخرية أو المزاح كان يدعو أتاري باسم حاكنتوش Jackintosh، نسبة لمؤسس الشركة Jack Tramiel، مشروع أتاري متوقف منذ عام 1993.

أتاري توس TOS 4.92 (هذا نسخة من MultiTOS في صيغة IMG. تكتب إلى رقاقة ROM). لكن نسخ 4.9x لم تصدر رسميا، رغم تسريب النماذج الأولية ومشاركتها لاحق، وأخر إصدار رسمي كان TOS V4.04

حاسوب أتاري أس تي، استخدام نظام تشغيل أتاري توس TOS المرتكز أساسا على نظام جيم دوس GEMDOS مع واجهة جيم GEM، ونسخة معدلة من نظام ملفات FAT12 على الأقراص المرنة ½ 3 (ميكرو فلوبي [31]) (أو FAT16 على الأقراص الثابتة). قبل إستخدام الأقراص الثانية في أجهزة الحاسوب المنزلي، كان TOS يشتغل من رقاقات ROM، وقبل ذلك في النسخ الأولى من أتاري أس تي، كان يقلع من الأقراص المرنة.

أهم اختلافات في قطاع الإقلاع بين DOS/TOS:

- قطاع الإقلاع لا يحتاج أن يتضمن متتالية القفزة المتوافقة مع IBM (أي 0xE9 xx xx أو 0xEB xx 90)،
- قطاعات الاقلاع التنفيذية في منصات أتاري تبدأ بشفرة تشغيل قفزة MC68K (مثل 0x603C) معالج موتورولا.
- القطاع يفتقد إلى لصيقة اسم صانعي القطع الأصلية OEM (قبل إصدار TOS 1.04) متوافقة مع أنظمة PC.
- للدلالة على قابلية قطاع الاقلاع على الإقلاع، يستخدم تدقيق المجموع في أخر القطاع (بينما نسخة PC تستخدم توقيع 0xAA 0x55 في الحاسوب الشخصي PC).
- خوارزمية تدقيق المجموع تعالج قيم 256 (نظام نهوي-كسر) في قطاع الاقلاع 512 بايت وتشمل 2 بايت الأخيرة، إذا كانت النتيجة الرقم السحري $1234 يصبح القطاع قابل للإقلاع.

على عكس م.س.دوس MS-DOS، نظام جيم دوس GEMDOS يسمح بالأقراص الاستثنائية متعددة المسار والقطاع، لذلك الأقراص التي تملك 10 أو حتى 11 قطاع لكل مسار وفوق 80 مسار مهيئة لم تكن نادرة في مجتمع أتاري. وعادة تستخدم 10 قطاعات في كل مسار من 80 مسار، ينتج عنها سعة 800 كيلوبايت غير مهيئة، لكن العديد من المستخدمين تجاوز السعة في الأقراص ذات الكثافة المزدوجة ووصل إلى 900 كيلوبايت باستخدام تهيئة خاصة.[7]. أنظمة ملفات قرص جيم دوس GEMDOS يمكن أيضا قرأتها باستخدام دوس أو ويندوز 9x .

قطاع إقلاع أتاري توس TOS

قطاع إقلاع TOS، في حاسوب أتاري أس تي، يقع في أول قرص منطقي (في القسم المعياري). بحجم قطاع منطقي واحد، حتى وإن تضمن قطاع الاقلاع المنطقي أكثر من قطاع فيزيائي، سوف تستخدم فقط 512 بايت الأولى، وبقية القطاع يتم حشوها (بايت 00).

نظام التشغيل TOS يقرأ هذا القطاع لإيجاد معلومات القرص المهمة. الكتلة المحملة BPB من هذا القطاع تخزن في بنية TOS. هذا القطاع يتضمن أيضا روتين للإقلاع النظام يسمح بتشغيل البرنامج الذي يقبل إعادة التموضع في الذاكرة زمن الاقلاع.

القسم المعياري في TOS

مواصفة أتاري AHDI 3.00 (مشغل للقرص الثابت) حددت نوعين من الأقسام المعيارية:

- القسم الاعتيادي (قسم GEM)
- القسم الكبير (قسم BGM)

بنية القسم المعياري نظام TOS

قطاع الاقلاع	محجوزة (اختيارية)	FAT #1	FAT #2	الدليل الجذر	منطقة البيانات للملفات والأدلة..(إلى نهاية القسم أو القرص)
(عدد القطاعات المحجوزة)		(عدد FATs) * (عدد القطاعات لكل FAT)		(عدد مدخلات الجذر * 32) / 512	عدد العناقيد * عدد القطاعات لكل عنقود

الحجم بعدد القطاعات. | في أتاري عند استخدام أقسام DOS&TOS عن طريق المشغلان PPDRIVER و HDDRIVER هذه البنية ستكون مختلفة قليلا.

أقسام DOS&TOS

ليس هناك معيار لأقسام DOS&TOS. فهي متوفرة فقط عن طريق مشغلات القرص PPDRIVER و HDDRIVER لكن مع إختلف في التطبيق. في هذا الفصل سنذكر بعض التفاصيل الفنية المستخدمة من قبل هذان المشغلان. معظم مشاكل عدم التوافق بين أنظمة ملفات FAT و TOS توجد في منطقة قطاع الاقلاع BPB. في الأسطر التالية وصف لتلك المعاملات الحرجة :

- المعاملان الأهم هما عدد بايتات لكل قطاع BPS وعدد القطاعات لكل عنقود SPC. كلهما يفسر بشكل مختلف في TOS و DOS/FAT لكنهما معا يحددان مفهوما للقطاعات المنطقية. في نظام ملفات TOS القطاع المنطقي (BPS) يمكن أن يمتد من 512 إلى 8192 بايت [5] و SPC دائما = 2..

  في نظام ملفات DOS/FAT، القطاع المنطقي (SPC * BPS) و BPS دائما = 512 بايت، لكن SPC يمكن أن يمتد من 2 إلى 128 الذي ينتج قطاع منطقي من 1024 إلى 65536 بايت.

  نستنتج أن النظامان يستخدمان تخطيطان مختلفان في تحديد القطاعات المنطقية الأكبر من 512 بايت. مثلا:
  - في نظام ملفات TOS، قطاع منطقي من 8192 بايت سيتحقق عن طريق BPS = 4096 و SPC = 2.
  - في نظام ملفات DOS، نفس القطاع المنطقي 8192 بايت سيتحقق عن طريق BPS = 512 و SPC = 16.
- المعامل الأخر المهم هو عدد القطاعات الإجمالي.
  - في نظام ملفات TOS يخزن هذا الرقم بقيم 16-بت (NSECTS) هذا ينتج حجم أقصى 512 ميغابايت (2^{16} * 8192 بايت) [6] من أجل قسم TOS.
  - في نظام ملفات DOS/FAT عدد القطاعات يمكن أن يخزن بقيم 32-بت (حقل HSECTS الحيد 020h في DOS 3.31 BPB)، هذا يسمح بحجم أقسام يصل إلى 2 تيرابايت.

بناء على ذلك، ولأن الجزء الخاص بجيم دوس GEMDOS في توس TOS لا يتعامل بالشكل الصحيح مع بعض DOS BPS يمكن فقط استخدام أقسام تصل إلى 32 ميغابايت (FAT16A) على أنظمة أتاري (ما لم يستخدم BigDOS كبديل لنظام GEMDOS).

لتجاوز هذا القيد المشغلان HDDRIVER و PPTDRIVER يقدمان نوع جديد من الأقسام يدعى TOS&DOS. كلا المشغلان يستخدم نفس التقنية لكن التطبيقات ستكون مختلفة.

أساسا الفكرة هي أن قسم TOS&DOS سيبدو مثل قسم TOS، مع قطاع لإقلاع TOS، عند استخدامه على أجهزة أتاري مع نظام TOS. نفس القسم سيبدو مثل قسم DOS، مع قطاع إقلاع DOS، عند النفاذ إليه من الحاسوب الشخصي الذي يستخدم نظام دوس/ويندوز. هذا يعني أن قسم TOS&DOS سوف يملك قطاعين للإقلاع؛ أحدهما من أجل TOS والأخر من أجل DOS.

بنية القسم في نظام DOS&TOS

قطاع إقلاع DOS	قطاع إقلاع TOS	FAT1	FAT2	الدليل الجذر	منطقة البيانات

لذلك القيود الموجودة في قسم TOS&DOS ستتبع قيود قسم TOS (العائق الأكبر). حجم القسم الأقصى يعتمد على إصدارة TOS، ومشغلات القرص الثابت، وسعة مهيئ المضيف host adapter . مع مشغلات القرص الحالية وجهاز host adapter[23]، (التي تدعم مجموعة الأوامر الممتدة/الموسعة extended command set من شركة ICD المصنعة للجهاز) حجم القسم الأقصى سيكون:

حجم	إصدارة
يصل إلى 256 ميغابايت	TOS < 1.4
يصل إلى 512 ميغابايت	TOS ≥ 1.4
يصل إلى 2 جيجابايت	فالكون (Falcon) TOS ≥ 4.x

أول قطاع على قرص ST سيكون مسار الاقلاع. الذي يخبر ST عن عدة أشياء ضرورية تخص القرص وعن إمكانية تحميل برنامج الاقلاع من القرص أو أن الشفرة ستكون في مكان أخر,

أولا، تدقيق المجموع $1234 (4660) في القطاع يعني أن القطاع قابل للإقلاع. إذا كان تدقيق المجموع صحيح، سوف يقفز النظام (قفزة طويلة) JSR إلى أول بايت في الصوان حيث تم تحميل شفرة الاقلاع. وبما أن موقع الصوان متغير، الشفرة في قطاع الاقلاع يجب أن تكون نسبية، وليست تابعة الموقع location-dependant.

تتم كتابة قطاع الاقلاع عادة عند تهيئة القرص أو نسخ كامل القرص إلى قرص أخر. قطاع الاقلاع بتضمن أيضا معاملات BPB الخاصة بالقرص، كما تظهر في الجدول التالي. أيضا في الجدول إذا ظهرت قيمة واحدة فقط، ستكون نفسها في جميع الأقراص الثلاثة، إذا ظهرت قيمتان ستكون متماثلة في قرص SS (ذو وجه واحد، بدون إقلاع) وقرص DS (ذو وجهين، بدون إقلاع).

أقراص أتاري أس تي المرنة المهيئة بنظام ملفات FAT تملك تخطيط مشابه جدا للقطاع الاقلاع في أنظمة مايكروسوفت/أي بي أم.

وصف	طول / بايت	رمز تذكري	إزاحة من بداية كتلة الاقلاع
تعليمة القفزة. 2 بايت تتضمن تعليمة قفزة إلى شفرة الاقلاع في قطاع إقلاع TOS، إذا كان القرص بقبل الاقلاع، و إلا لن تستخدم. قطاعات الاقلاع الأصلية في أتاري أس تي (معالج موتورولا 68000) تبدأ بالقفزة القصيرة ??0x. 0x60 وللتوافق مع أنظمة PC الأقراص المهيئة في أتاري أس تي، منذ 1.4 TOS تبدأ بالتعليمة ??0x. x0E9 قفزة (أتاري) Original Atari ST/ 68000 BRA.S 0x60 | 0x?? | (empty) قفزة (أتاري للتوافق) Atari ST / compatibility with PC 0xE9 | 0x?? | (empty) حرف S بعد BRA، ترشد المجمع Assembler إلى أن التعليمة هي قفزة قصيرة short branch. كلمة TOS اختصار لاسم نظام التشغيل المستخدم في حاسوب أتاري أس تي، مع معالج موتورولا 68000. قرص ذو وجه واحد، بدون إقلاع = SS/ قرص ذو وجهين، بدون إقلاع = DS. قرص إقلاع TOS = BOOT DS | SS | BOOT | [24] (empty) | 00 00 | 60 38 | LSB/MSB	2	BRA.S	0000h (0)
لصيقة اسم صانعي القطع الأصلية OEM (قد تكون محشو بفراغات؛ بايت 0x20)، في المثال لصيقة على وحدات التخزين التي تتضمن محمل إقلاع أتاري. راجع أعلاه OEM لحظ اختلف الازاحة والطول مقارنة بالمدخلة على الأقراص المهيئة في PC. أسكي | ست عشري "Loader" | 4C 6F 61 64 65 72	6	OEM	0002h (2)
الرقم التسلسلي للقرص (24 بت) (الاعتيادية : 0x00 0x00 0x00). يستخدمها أتاري أس تي للكشف عن تغيير القرص. (متعقب وحدة تخزين ويندوز 98/95 سيخزن دائما "IHC" هنا على الأقراص المرنة الغير المحمية من الكتابة) هذه القيمة يجب أن تتغير إذا تغير محتوى القرص خارجيا، وإلا أتاري أس تي لن يتعرف على التغيير عند إعادة إدراج القرص. هذه المدخلة تغطي على حقل OEM على الأقراص المهيئة في PCوللحصول على أقصى توافق، يجب هنا تطابق بعض الأنماط: انظر أعلاه.	3	SERIAL	0008h (8)
كتلة أتاري BPB. (القرص المرن) محتوى | بايت | رمز تذكري | إزاحة DOS 2.0 BPB **حجم القطاع** (2 بايت بنية إنتيل) عدد بايتات في القطاع المنطقي. (نهوي-كبير) **عدد القطاعات في كل عنقود.** يجب أن يكون قوة العدد 2. نظام GEMDOS يدعم فقط 2. **القطاعات المحجوزة** (2 بايت بنية إنتيل) عدد القطاعات المنطقية المحجوزة في بداية القرص المنطقي، وتشمل قطاع الاقلاع. عادة تكون 1 (في FAT12/FAT16). **عدد FATs.** عدد نسخ FATs المخزنة على القرص المنطقي. عادة تكون 2. **عدد المدخلات الإجمالي في الدليل الجذر** (2 بايت بنية إنتيل) العدد الإجمالي لمدخلات أسماء الملفات التي يمكن تخزينها في الدليل الجذر للقرص المنطقي. **عدد القطاعات الإجمالي على القرص المنطقي** (2 بايت بنية إنتيل) العدد الإجمالي للقطاعات المنطقية على القرص المنطقي وتشمل القطاعات المحجوزة. **واصف الوسيط** بايت واصف الوسيط سيكون F8 على الأقراص الثابتة. لكن لا يستخدم من قبل نظام ST BIOS. (قارن مع FAT ID) **عدد القطاعات المنطقية لكل FAT** (هذه 2 بايت بنية إنتيل) عدد القطاعات المنطقية التي تحتلها كل نسخة من FAT. DOS 3.0 BPB **عدد القطاعات الفيزيائية في كل مسار** (2 بايت بنية إنتيل) غير قابل للتطبيق على القرص الثابت. المدخلة 0، تشير إلى أن المدخلة محجوزة، لكن ليس مستخدمة. **عدد الرؤوس/الجوانب على الوسيط** (2 بايت بنية إنتيل) غير قابل للتطبيق على القرص الثابت. المدخلة 0، تشير إلى أن المدخلة محجوزة، لكن ليست مستخدمة. **عدد القطاعات المخفية** (2 بايت بنية إنتيل) لا يجب استخدمها إذا كانت مدخلة القطاعات المنطقية 0 عند 0x13. وهذا لا تتوافق مع 0x01C في DOS 3.31 BPB DS | SS | BOOT | LSB/MSB 00Bh (11) | BPS | 2 | (empty) | 00 02 00Dh (13) | SPC | 1 | (empty) | 02 00Eh (14) | RES | 2 | (empty) | 01 00 010h (16) | NFATS | 1 | (empty) | 02 011h (17) | NDIRS | 2 | (empty) | 70 00 013h (19) | NSECTS | 2 | (empty) | D0 02 | D0 02 | A0 015h (21) | MEDIA | 1 | (empty) | F8 | F8 | F9 016h (22) | SPF | 2 | (empty) | 05 00 018h (24) | SPT | 2 | (empty) | 09 00 01Ah (26) | NSIDES | 2 | (empty) | 01 00 | 01 00 | 00 02 01Ch (28) | NHID | 2 | (empty) | 00	19	BPB	00Bh (11)
بيانات قطاع إقلاع خاصة (بنية مختلطة نهوي-صغير و نهوي-كبير) / شفرة إقلاع (إن وجدت)	متفاوت		01Eh (30)
شفرة الاقلاع. شفرة إقلاع تخص نظام ملفات وتشغيل أتاري أس تي. وليس هناك أية فرضيات لموقع تحميل الشفرة، التي يجب أن تقبل إعادة التعين، إذا فشل تحميل النظام، يمكن للشفرة أتاري العودة إلى BIOS مع تعليمة RTS 68000 (شفرة التشغيل 0x4E75 مع متتالية 0x75 0x4E نهوي-كبير) وجميع التسجيلات بدون تعديل	متفاوت		متفاوت

تدقيق المجموع، 2 بايت الأخيرة في قطاع الاقلاع محجوزة من أجل تدقيق المجموع 16-بت. الذي يجب أن يساوي $1234. (4660) في جهاز أتاري أس تي 68000، خوارزمية تدقيق المجموع تضيف 256 كلمة نهوي-كبير وتشمل 2 بايت هذه في قطاع الاقلاع 512 بايت، إذا كانت النتيجة تساوي الرقم السحري $1234. نظام BIOS سيعتبر هذا القطاع قابلا للإقلاع. مدخلة تدقيق المجموع هذه يمكن استخدمها من أجل محاذاة تدقيق المجموع وفقا لذلك. إذا كان حجم القطاع المنطقي أكبر من 512 بايت، لا يتم تضمين بقية القطاع في تدقيق المجموع، وتصفر. وبما أن بعض أنظمة تشغيل PC، لا تقبل بالخطأ الأقراص المرنة المهيئة بنظام FAT إذا كان توقيع القطاع 0x55 0xAA غير موجود هنا، ينصح بوضع التوقيع 0x55 0xAA في هذا المكان (وإضافة محمل إقلاع أو روتين وهمي متوافق مع أنظمة IBM) واستعمال 2 بايت غير مستخدمة في منطقة شفرة الاقلاع أو البيانات الخاصة أو الرقم التسلسلي لتتأكد من أن تدقيق المجموع 0x1234 لا يتطابق (إلا إذا كان ملف أو غطاء fat code المشترك تنفيذي في أتاري أس تي وفي IBM PC في نفس الوقت). مدخلة تدقيق المجموع في قطاعات إقلاع Atari تحتفظ بقيمة المحاذاة، وليست قيمة الرقم السحري نفسه. قيمة الرقم 0x1234 لا تخزن في أي مكان على القرص. مقارنة بمعالجات إنتل x86، معالجات موتورولا 680x0 المستخدمة في أجهزة أتاري تستخدم نهوي-كبير ويجب أخذ هذا التمثيل بالاعتبار عند حساب تدقيق المجموع. ونتيجة لهذا من أجل شفرة التحقق من صحة تدقيق المجموع التي تشتغل على أجهزة x86، يجب تبديل أزواج بايتات قبل إضافة 16-بت. • لأن الأجهزة المتوافقة مع أنظمة IBM تستخدم نهوي صغير في قطاعات الإقلاع، في التوقيع عند الحيد 1FEh سيكون 55h عند 1FEh و AAh عند 1FFh ويمكن كتابتها بكلمة 16-بت AA55h في برامج المعالج x86 ، بينما تكتب 55AAh في برامج المعالجات الأخرى بطريقة نهوي كبير. • روتين وهمي = stub / dummy routine، و Word = الكلمة أي 2 بايت أو 16-بت • الشفرة الثنائية FAT Binary /FAT Code : ملف تنفيذي يتضمن شفرة لأكثر من معالج واحد. ويتم اختيار الشفرة الصحيحة آليا زمن التشغيل. • سجل الاقلاع الرئيسي MBR في أتاري يدعى قطاع الجذر Root Sector. ويستخدم أيضا تدقيق المجموع $1234.	2	01FEh (510)

بنية قطاع إقلاع أتاري أس تي (قطاع إقلاع القرص المرن)

- تعليمة القفزة (2 بايت) تقفز إلى شفرة الاقلاع عند الحيد $3A (أي 38h بايت للأمام)
- لصيقة الاسم على وحدات التخزين التي تتضمن محمل إقلاع أتاري أس تي (6 بايت) ؛ سلسلة المحارف 'Loader'
- الرقم التسلسلي للقرص (3 بايت)

منطقة معاملات BPB

- عدد بايتات في كل قطاع (2 بايت)
- عدد القطاعات في كل نسخة FAT (2 بايت)
- عدد القطاعات في كل عنقود (1 بايت)
- عدد القطاعات لكل مسار (2 بايت)
- عدد القطاعات المحجوزة (2 بايت)
- عدد الجوانب على الوسيط (2 بايت)
- عدد نسخ FAT في نظام الملفات (1 بايت)
- عدد القطاعات المخفية (2 بايت)
- عدد مدخلات الجذر (2 بايت)
- بيانات قطاع إقلاع خاصة (مختلطة نهوي-صغير ونهوي-كبير متفاوت)
- عدد القطاعات على الوسيط (2 بايت)
- محجوزة (1 بايت)
- واصف الوسيط (1 بايت)

منطقة الشفرة

- شفرة إقلاع، (محشوة ببايت 00 من عند النهاية لملء كامل القطاع)
- تدقيق المجموع (2 بايت)؛ يتم حساب هذه 2 بايت حتى يصبح تدقيق المجموع الرقم السحري $1234 ؛ ويعتبر القطاع قابل للإقلاع.

(القطاع المنطقي 0) CHS 0-0-1

	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	0123456789ABCDEF
0000	60	38	4C	6F	61	64	65	72	00	00	00	00	02	02	01	00	[`8Loader........]
0010	02	10	00	24	03	F8	02	00	0A	00	01	00	00	00	00	00	[...$.ø..........]
0020	00	00	00	00	00	00	00	00	00	00	00	00	00	00	53	57	[..............SW]
0030	4F	4F	53	48	20	20	49	4D	47	00	70	07	32	3C	07	77	[OOSH IMG.p.2<.w]
0040	48	E7	C0	00	3F	3C	00	25	4E	4E	54	8F	4C	DF	00	03	[HçÀ.?<.%NNT.Lß..]
0050	20	7C	FF	FF	82	40	30	81	04	41	01	11	51	C8	FF	E2	[|ÿÿ‚@0..A..QÈÿâ]
0060	20	7C	FF	FF	82	40	70	0F	42	58	51	C8	FF	FC	33	FA	[|ÿÿ‚@p.BXQÈÿü3ú]
0070	FF	AE	00	00	04	82	3F	39	00	00	04	46	3F	3C	00	07	[ÿ®...‚?9...F?<..]
0080	4E	4D	58	4F	4A	80	67	00	00	F6	2A	40	41	FA	FF	9C	[NMXOJ€g..ö*@Aúÿœ]
0090	4A	90	66	06	20	B9	00	00	04	32	30	2D	00	08	E1	48	[J.f. ¹...20-..áH]
00A0	D0	80	38	40	D9	FA	FF	84	30	3A	FF	76	67	10	3C	3A	[Ð€8@ÙúÿH0:ÿvg.<:]
00B0	FF	72	38	3A	FF	70	26	7A	FF	6E	60	00	00	B4	3C	2D	[ÿr8:ÿp&zÿn`..´<-]
00C0	00	0A	38	2D	00	08	D8	6D	00	06	26	7A	FF	5E	61	00	[..8-..Øm..&zÿ^a.]
00D0	00	B0	66	00	00	AA	20	4C	30	2D	00	06	E1	48	E3	48	[.°f..ª L0-..áHãH]
00E0	41	F0	00	00	43	FA	FF	48	90	FC	00	20	B1	CC	6D	00	[Að..CúÿH.ü. ±Ìm.]
00F0	00	8E	70	0A	12	30	00	00	B2	31	00	00	66	EA	51	C8	[.Žp..0..²1..fêQÈ]
0100	FF	F4	7E	00	1E	28	00	1B	E1	4F	1E	28	00	1A	2C	7A	[ÿô~..(..áO.(..,z]
0110	FF	1A	26	7A	FF	12	42	84	0C	47	0F	F0	6C	52	36	07	[ÿ.&zÿ.B„.G.ðlR6.]
0120	55	43	C6	ED	00	02	D6	6D	00	0C	0C	44	00	40	6C	08	[UCÆí..Öm...D.@l.]
0130	4A	44	67	0E	B6	45	67	10	61	46	66	42	E1	8C	E3	8C	[JDg.¶Eg.aFfBáŒãŒ]
0140	D7	C4	3C	03	3A	03	42	84	D8	6D	00	02	DA	6D	00	02	[×Ä<.:.B„Øm..Úm..]
0150	34	07	E2	4A	D4	47	12	36	20	01	E1	49	12	36	20	00	[4.âJÔG.6 .áI.6 .]
0160	08	07	00	00	67	02	E8	49	02	41	0F	FF	3E	01	60	A8	[....g.èI.A.ÿ>.`¨]
0170	4A	44	67	04	61	0A	66	06	2F	3A	FE	AC	4E	75	60	FE	[JDg.a.f./:þ¬Nu`þ]
0180	3F	39	00	00	04	46	3F	06	3F	04	2F	0B	42	67	3F	3C	[?9...F?.?./.Bg?<]
0190	00	04	4E	4D	DE	FC	00	0E	4A	40	4E	75	00	00	00	00	[..NMÞü..J@Nu....]
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	24	7D	[..............$}]
	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	

رغم أن بعض أنظمة الملفات تستخدمه، نظام TOS لا يستخدم بايت واصف الوسيط. هذا النظام يتجاهل أيضا عدد القطاعات المخفية على الأقراص المرنة. بايتات OEM تستخدم على قرص الاقلاع وربما كذلك على أقراص الشركات الأخرى، لكنها عموما لا تستخدم على الأقراص التي لا تقبل الاقلاع. الرقم التسلسلي يكتب في زمن التهيئة ويجب أن يكون فريد كي يستطيع TOS معرفة القرص إذا تبدل. بعض الإداوات كي تستطيع التعامل مع المحمل، يجب أن يكون OEM بالشكل'Loader' و 2 بايت الأخيرة محفوظة لقيمة المعادلة التي تسمح لتدقيق المجموع أن يكون صحيح.

إزاحة	بايت	اسم	حقول بيانات منطقة قطاع الاقلاع الخاصة (محمل الاقلاع) بترتيب مختلط نهوي-صغير ونهوي-كبير
01E (30)	2	EXECFLG	علم تنفيذ، قيمة 2 بايت (word) المنسوخة إلى متغير النظام الذي يدعى '_cmdload' (بترتيب 2-بايت نهوي كبير) .
020 (32)	2	LDMODE	نمط التحميل Load mode (ترتيب 2 بايت نهوي كبير). عند الحيد 020. • القيمة صفر 0 تأمر المحمل بالبحث عن وتحميل ملف FNAME عن طريق اسمه واستخدام نظام الملفات. • القيم الأخرى تأمر المحمل بتحميل سلسلة من القطاعات في SCETCNT بداية من SSECT دون الإشارة إلى نظام الملفات.
022 (34)	2	SSECT	أول قطاع يقرأ (نهوي كبير). يستخدم فقط عندما لا تكون القيمة 0 في نمط التحميل.
024 (36)	2	SCETCNT	عدد القطاعات التي ستحمل / ستقرأ (نهوي كبير). يستخدم فقط عندما لا تكون القيمة 0 في نمط التحميل.
026 (38)	4	LDADDR	عنوان التحميل (2 بايت نهوي كبير). عنوان الذاكرة حيث سيتم تحميل الملف أو القطاعات
02A (42)	4	FATBUF	عنوان FAT ـ (4 بايت نهوي كبير) يشير إلى عنوان الذاكرة أين سيتم تحميل بيانات FAT والدليل الجذر. القيمة 00000000$ ستختار آليا العنوان المناسب.
02E (46)	11	FNAME	اسم الملف و امتداد (اسم 8.3 بدون '.') من 8 محارف + 3 في امتداد. يستخدم فقط عندما تكون القيمة 0 في نمط التحميل 53 57 4F 4F 53 48 20 20 49 4D 47 'SWOOSH IMG'
039 (57)		reserved	محجوزة.
03A (58)		BOOTIT	شفرة إقلاع

عدد المدخلات في الدليل الجذر (2 بايت نهوي صغير). كل مدخلة 32 بايت * 16 مدخلة = قطاع 512 بايت. محمل إقلاع ST يمكنه تحميل ملف 'image file' من أي قرص بصرف النظر عن مكان ظهوره في الدليل

حاسوب أم أس أكس MSX

(SONY MSX MSX-DOS/MSX BASIC)

هذا الحاسوب المنزلي أعلنت عنه مايكروسوفت في 16 يونيو/حزيران عام 1983. المسؤول عن مشروع MSX كان يدعى Kazuhiko Nishi، ثم نائب مدير مايكروسوفت اليابان والمدير في شركة أسكي ASCII التي صممت الجهاز بالتعاون مع مايكروسوفت. هذه الأخيرة وفرت البرنامج الثابت للغة بيسيك BASIC ؛ (لغة البرمجة الممتدة من مايكروسوفت بيسيك Microsoft Basic).

رغم مشاركة مايكروسوفت في المشروع، إلا أن جهاز MSX كان نادرا في الولايات المتحدة، لكنه كان معروف في معظم اليابان، وفي الشرق الأوسط، والبرازيل، والاتحاد السوفياتي، وهولندة، وإسبانيا، وأقل انتشار في عدة دول أوربية أخرى. من الصعب تقدير عدد الأجهزة المبيعة حول العالم من MSX، لكن في اليابان تقدر بخمس مليون وحدة.

النسخة العربية من هذه الأجهزة كانت في الثمانينات من صخر. بينما الطراز Sakhr MSX AX170 كان الأكثر شعبية في البلاد العربية.

مواصفة الجهاز (حسب الطراز)
الأجهزة: MSX/MSX2/MSX2+/MSX turboR
معالج ميكرو 8-بت: Zilog Z80،
ذاكرة وصول عشوائي RAM : من 8-512 كيلوبايت.
ذاكرة للقراءة فقط ROM : حجم متفاوت 32/48/64/96 كيلوبايت.
شريحة ROM: تضمنت البرنامج الثابت للغة بيسيك MSX BASIC ونظام BIOS (ولاحقا تضمنت إضافات أخرى مثل DiskROM.....)
ذاكرة للفيديو VRAM: حجم متفاوت 16/128/192/ كيلوبايت. وشريحة للصوت.
قرص مرن 3.5 بوصة (باستثناء الطراز الأول).
نظام التشغيل MSX-DOS إلى جانب لغة البرمجة MSX BASIC وكلهما من شركة مايكروسوفت.

الأحرف المختصرة MSX في الغالب، تعني جملة مايكروسوفت (بيسيك) الممتدة "MicroSoft eXtended". قيل أيضا أن المشروع تزعمته مايكروسوفت كمحاولة لتوحيد المعايير وسط مصنعي العتاد أنذاك. جهاز MSX أنتجته شركات عدة مثل، سوني Sony، توشيبا Toshiba، و باناسونيك Panasonic ، دايو Daewoo، و فيليبس Philips.

نظام التشغيل أم أس أكس دوس MSX–DOS

النظام الرسمي الوحيد المستخدم في أجهزة MSX كان يدعى أم أس أكس دوس MSX-DOS وكان مزيج بين مايكروسوفت دوس MS-DOS 1.25 [8] ونظام CP/M-80 2. وأخر إصدارته كانت 2.31 في عام 1990 مع حاسوب MSX Turbo-R.

الإصدارة MSX-DOS 2 [9] تضمنت ذاكرة ROM ونواة (مع بعض امتدادات لغة بسيك) وملفات نظام التشغيل COMMAND2.COM و MSXDOS2.SYS. على القرص، حيث يوجد الكثير من الملفات المساعدة وبعض الإداوات. لكن MSX-DOS 2 تبدوا ظهريا أكثر تعقيدا من إصدارة MSX-DOS 1.0x.

في زمن تطوير نظام MSX-DOS كان الخيار الوحيد في الجهاز للتخزين البيانات الكبيرة هو القرص المرن، النظام كان يعمل جيدا "كنظام تشغيل قرص مرن"، لكن مع الوقت ظهرت خيارات للتخزين أخرى في شكل عتاد للهوة مركب (في بداية التسعينات كانت في شكل متحكمات القرص الثابت SCSI و IDE وفي عصرنا، أجهزة قراءة بطاقات الوسائط المتعددة). نظام MSX–DOS استخدم في إدارة هذه الأجهزة، لكن كانت هناك بعض المشاكل:

- نظام MSX-DOS يتعامل مع أعداد القطاع بقيم 16-بت، ويدعم فقط نظام ملفات FAT12. هذا يحد من حجم وحدة التخزين التي لا يمكنها أن تتجاوز 32 ميغابايت. لكن كانت هناك رقع patches غير رسمية لدعم نظام ملفات FAT16 .(الرقعة هي إضافة مؤقتة إلى جزء من شفرة البرنامج)
- مشغل العتاد الفعلي (الشفرة التي تتعامل مع عتاد متحكمات التخزين الكبيرة) مضمن داخل ذكرة ROM الخاصة بنواة نظام التشغيل، وممثل في الحاسوب عن طريق متحكمات القرص المرن الخارجية ومحركات القرص المرن المدمجة. ولا توجد طريقة رسمية موثقة للتضمين مشغل عتاد مخصص في ذاكرة النواة ROM ؛ ولذلك تحتم على مطوري هذا العتاد الخاص (لمتحكمات التخزين) عمل برمجة عكسية (reverse-engineer) لشفرة النواة لتضمين المشغل المخصص custom driver.
- هناك علاقة واحد-لواحد مباشرة وثابتة بين محارف القرص كما تظهر للمستخدم ووحدات الجهاز التي تعرضها واجهة برمجة التطبيقات API لمشغل العتاد. على سبيل المثال، للنفاذ أو للوصول إلى القرص :A، نظام MSX-DOS يطلب من المشغل النفاذ إلى أول جهاز فيه ؛ بينما يستعلم عن الجهاز الثاني عند النفاذ إلى القرص :B. هذا لا يشكل مشكلة مع الأقراص المرنة، لكن عند استخدام أجهزة أكثر تعقيد تملك قسم أو عدة أقسام، المشغل سيكون المسؤول (وعادة الأدوات الخارجية التي من صنع مطور المشغل ستكون أيضا مسؤولة) عن إدارة إسناد القرص إلى الجهاز والقسم.
- أما إدارة الأجهزة التي لا تملك كتل non-block (مثل الأقراص المدمجة CD-ROM) فصعب جدا، ويحتاج إلى عمل برمجة عكسية على شفرة النواة.

إقلاع MSX-DOS

عكس MS-DOS 2.x، نظام MSX-DOS استخدم روتينيات BDOS ROM في عملية الاقلاع ولم يستخدم قطاع الاقلاع على القرص المرن في الاقلاع، لكنه مثل MS-DOS 1.25 استخدم قيمة هوية FAT ID من أول بايت في نسخة FAT لاختيار معامل نظام الملفات FAT12 بدلا من العودة إلى كتلة BPB في قطاع الاقلاع.

ورغم أنه لم يوفر طريقة للإقلاع القرص المرن!. لكن النظام يستطيع الاقلاع من عدة محركات أقراص مرنة مع وجود أكثر من حاوية (خرطوشه) قرص مرن FDC في أكثر من منفذ للقرص، (أي إمكانية وجود محرك القرص المرن ¼ 3 إلى جانب ½ 3، وإمكانية استخدام أحدهما مع قرص قابل للإقلاع).

رغم معيارية بنية القرص المستخدمة في MSX-DOS 1 و MSX-DOS 2، إلا أن MSX-DOS 1 لا يستخدم المعلومات المخزنة في مناطق معينة في القرص (قطاع الاقلاع)، لذلك هذه المعلومات ليس صحيحة بالضرورة على أقراص MSX-DOS 1 ويمكن أن تتسبب في مشاكل عند استخدام MSX-DOS 2 مع هذه الأقراص. أيضا، الأمر UNDEL سوف يعمل فقط مع الأقراص المهيئة في MSX-DOS 2 (أي، الأقراص التي تملك الهوية " volume id" في قطاع الاقلاع) ولن يعمل مع أقراص MSX-DOS 1 أو الأقراص المهيئة في الأنظمة أخرى.

في MSX-DOS، القطاعات موزعة على أربعة مناطق على القرص، (أنظر للجدول)، بيانات الملفات التي تكتب إلى القرص تسجل في منطقة البيانات. ومعلومات معالجة البيانات في ثلاثة مناطق. قطاع الاقلاع سيكون دائما في القطاع 0، لكن القطاعات العليا الأخرى (FAT، والدليل، ومنطقة البيانات) ستكون مختلفة وفقا للوسيط، لهذا يجب العودة إلى جدول DPB.

جدول كتلة معاملات القرص DPB وقطاع الاقلاع

في MSX–DOS، يتم تخصيص DPB في منطقة العمل في الذاكرة لكل قرص متصل، حيث تسجل معلومات القرص. MSX–DOS يمكنه التعامل مع معظم أنوع محركات الأقراص، لأن الاختلاف بين الوسائط يمكن معادلته عن طريق معالج تمثيل الأقراص.

المعلومات المكتوب في جدول DPB، والتي هي في الأصل على قطاع الاقلاع على القرص (القطاع #0)، تقرأ عند بدء تشغيل MSX–DOS.لحظ الاختلاف بين محتويات قطاع الاقلاع وجدول DPB، كما تظهر في معاملات BPB 3.0 والجدول التالي. البيانات ستكون مرتبة بشكل مختلف في قطاع الاقلاع وجدول DPB.

	9 قطاعات/1DD	9 قطاعات/2DD	8 قطاعات/1DD	8 قطاعات/2DD
نوع الوسيط	0F8H	0F9H	0FAH	0FBH
عدد الجوانب	1	2	1	2
عدد مسارات كل جانب	80	80	80	80
عدد بايتات كل مسار	9	9	8	8
عدد بايتات كل قطاع	512	512	512	512
حجم العنقود (بالقطاع)	2	2	2	2
حجم FAT (بالقطاع)	2	3	1	2
عدد نسخ FAT	2	2	2	2
عدد الملفات القابلة للتسجيل	112	112	112	112

الوسيط المدعوم في MSX-DOS

رقم القرص		قاعدة ←
واصف الوسيط (هوية)	Media ID	+1
حجم القطاع		+2 +3
قناع الدليل	Directory mask	+4
إزاحة الدليل (تبديل)	Directory shift	+5
قناع العنقود	Cluster mask	+6
إزاحة العنقود (تبديل)	Cluster shift	+7
القطاع الأعلى للجدول توزيع الملفات FAT		+8 +9
عدد نسخ FAT		+10
عدد مدخلات الدليل		+11
القطاع الأعلى لمنطقة البيانات		+12 +13
كمية العناقيد +1		+14 +15
عدد القطاعات المطلوبة من أجل نسخة FAT		+16
القطاع الأعلى لمنطقة الدليل		+17 +18
عنوان FAT في الذاكرة		+19 +20

تستخدم وظيفة نداء النظام 1BH للنفاذ إلى DPB. (تحصيل معلومات القرص) النداء يعود بعنوان DPB في الذاكرة ومعلومات أخرى لكل قرص كتب على قطاع الاقلاع.

MSX-DOS يدعم حتى 8 أقراص. في نظام محرك الأقراص الواحد, ويملك ميزة محاكاة قرصيين (عبر استبدال الأقراص المرن مؤقتا), ويدعم استخدام لوحة المفاتيح, والشاشة, والطابعة.

النظام يملك مدير ملفات مرن لا يعتمد على البنية الفيزيائية للقرص, ويدعم عدة وسائط, يمكن استخدام نوعان من الأقراص المرنة المعيارية ذات الكثافة المزدوجة؛ وجه واحد 1DD أو وجهين 2DD. وكلاهما يستخدم بنية المسار بـ 8-قطاعات هذا يعني إمكانية استخدام أربعة أنواع من الوسائط. الجدول أعلاه يعرضها في مايكروسوفت

روتين تحديد نوع الوسيط

1. أقرأ قطاع الاقلاع (المسار 0، القطاع 1) من القرص المستهدف.
2. تأكد ما إذا كان البايت الأول هو 0E9h أو 0EBh (هذه هي تعليمة القفزة JMP في أنظمة 8086)
3. إذا فشلت الخطوة (2)، اعتبر أن إصدارة القرص قبل 2.0 MS-DOS ؛ لذلك، استخدم أول بايت من نسخة FAT يمرره المتصل وتأكد أنه بين 0F8h و 0FFh.
   1. إذا نجحت الخطوة (3)، استخدم هذا كواصف وسيط. إذا فشلت الخطوة (3)، إذن لا يمكن قراءة هذا القرص.
4. إذا نجحت الخطوة (2)، أقرأ بايتات من 0B # إلى 1D #. هذا DPB الخاص بنظام MS-DOS، الإصدارة 2.0 وما فوق. DPB الخاص بنظام MSX-DOS يمكن الحصول عليه من قطاع إقلاع MS-DOS (راجع الجدول التالي : قطاع الاقلاع في وحدات تخزين دوس MSX-DOS). لمعلومات أكثر، راجع الدليل الرسمي، باللغة الانجليزية، على الأنترنت.

وحدات تخزين MSX-DOS المهيئة بنظام ملفات FAT12 تملك تخطيط شبيه جدا بقطاع إقلاع دوس:

<table>
<tr><th>وصف</th><th>طول / بايت</th><th>مثال/رمز تذكري</th><th>إزاحة</th></tr>
<tr><td>تعليمة قفزة وهمية MSX-DOS / Dummy jump (مثال، 0xEB 0xFE 0x90)</td><td>3</td><td>EB FE 90</td><td>0x000</td></tr>
<tr><td>اسم صانعي القطع الأصلية OEM (يمكن أن يكون محشو بفراغات ؛ بايت 0x20).</td><td>8</td><td>SANYO2.0</td><td>0x003</td></tr>
<tr><td>كتلة DOS 3.0 BPB (تشمل معاملات DOS 2.0 BPB)
<table>
<tr><th>محتوى</th><th></th><th></th><th>بايت</th><th>إزاحة</th></tr>
<tr><td colspan="5">معاملات DOS 2.0 BPB</td></tr>
<tr><td rowspan="2">عدد بايتات في القطاع المنطقي. (حجم القطاع) قوة العدد اثنين، عادة تكون القيمة 512 بايت</td><td>0B</td><td>LSB [24]</td><td rowspan="2">2</td><td rowspan="2">00Bh (11)</td></tr>
<tr><td>0C</td><td>MSB [24]</td></tr>
<tr><td>عدد القطاعات المنطقية في كل عنقود. (حجم العنقود)، القيم المعترف بها هي: 1، 2، 4، 8، 16، 32، 64، 128</td><td colspan="2"></td><td>1</td><td>00Dh (13)</td></tr>
<tr><td rowspan="2">عدد القطاعات المنطقية المحجوزة. (عدد القطاعات الغير مستخدمة من قبل MSX-DOS) قبل أول FAT في صورة نظام الملفات. ستكون 1 على الأقل من أجل هذا القطاع. (وعادة 32 من أجل FAT32)</td><td>0E</td><td>LSB</td><td rowspan="2">2</td><td rowspan="2">00Eh (14)</td></tr>
<tr><td>0F</td><td>MSB</td></tr>
<tr><td>عدد نسخ FATs المخزنة على القرص المنطقي، تقريبا دائما 2</td><td colspan="2"></td><td>1</td><td>010h (16)</td></tr>
<tr><td rowspan="2">عدد المدخلات الإجمالي في الدليل الجذر. (أي عدد الملفات الممكن إنشائها) في FAT12 أو FAT16 لكن في FAT32 ستكون 0</td><td>11</td><td>LSB</td><td rowspan="2">2</td><td rowspan="2">011h (17)</td></tr>
<tr><td>12</td><td>MSB</td></tr>
<tr><td rowspan="2">عدد القطاعات الإجمالي على القرص المنطقي. وتشمل القطاعات المحجوزة. (عدد القطاعات لكل قرص) إذا كانت صفر، تستخدم قيمة 4 بايت عند الحيد 0x020</td><td>13</td><td>LSB</td><td rowspan="2">2</td><td rowspan="2">013h (19)</td></tr>
<tr><td>14</td><td>MSB</td></tr>
<tr><td>واصف الوسيط. (قارن مع FAT ID)</td><td colspan="2"></td><td>1</td><td>015h (21)</td></tr>
<tr><td rowspan="2">عدد القطاعات المنطقية في كل FAT. (حجم FAT) من أجل FAT12/FAT16. لكن في FAT32 تعين إلى 0 وتستخدم قيمة 32-بت عند الحيد 0x024 عوض ذلك.</td><td>16</td><td>LSB</td><td rowspan="2">2</td><td rowspan="2">016h (22)</td></tr>
<tr><td>17</td><td>MSB</td></tr>
<tr><td colspan="5">معاملات DOS 3.0 BPB</td></tr>
<tr><td rowspan="2">عدد القطاعات الفيزيائية في كل مسار PST. للأقراص التي تستخدم قياسات CHS INT 13h. مثلا 15 للقرص المرن 1.20 ميغابايت. المدخلة 0، تشير إلى أن المدخلة محجوزة، لكن ليس مستخدمة.</td><td>18</td><td>LSB</td><td rowspan="2">2</td><td rowspan="2">018h (24)</td></tr>
<tr><td>19</td><td>MSB</td></tr>
<tr><td rowspan="2">عدد الرؤوس على الوسيط (عدد الجوانب المستخدمة) للأقراص التي تستخدم قياسات CHS INT 13h. مثلا 2 للقرص المرن ذو وجهين. المدخلة 0، تشير إلى أن المدخلة محجوزة، لكن ليس مستخدمة.</td><td>1A</td><td>LSB</td><td rowspan="2">2</td><td rowspan="2">01Ah (26)</td></tr>
<tr><td>1B</td><td>MSB</td></tr>
<tr><td rowspan="2">عدد القطاعات المخفية التي تسبق وحدة التخزين FAT هذه. لا يجب استخدمها إذا كانت المدخلة 0 عند 0x13. وهذه لا تتوافق مع 0x01C في DOS 3.31 BPB ويجب أن تكون 0 على الوسيط غير المقسم.</td><td>1C</td><td>LSB</td><td rowspan="2">2</td><td rowspan="2">01Ch (28)</td></tr>
<tr><td>1D</td><td>MSB</td></tr>
</table></td><td>19</td><td></td><td>0x00B</td></tr>
<tr><td>مدخلة شفرة MSX-DOS 1 من أجل معالجات Z80 داخل شفرة إقلاع MSX. هذا الموقع حيث تقفز أجهزة MSX-DOS 1 عند تمرير التحكم إلى قطاع الاقلاع.
هذه الموقع أصبح يتداخل مع بنية BPB منذ DOS 3.2 أو شفرة قطاع الاقلاع المتوافقة مع x86 في قطاعات الاقلاع المتوافقة مع IBM PC، وسيؤدي إلى انهيار جهاز MSX ما لم تتخذ إجراءات وقائية خاصة كمثل وقف المعالج هنا في متكررة حلقية tight loop (شفرة التشغيل [4] JR 0x01E for 0xFE 0x18).</td><td>متفاوت (2)</td><td></td><td>0x01E</td></tr>
<tr><td>توقيع "VOL_ID" في وحدة تخزين MSX-DOS 2</td><td>6</td><td>VOL_ID</td><td>0x020</td></tr>
<tr><td>undelete flag علم استرجاع الملفات التي تم حذفها في MSX-DOS 2 (القيمة الاعتيادية: 0x00، إذا كان التوقيع "VOL_ID" موجود عند حيد القطاع 0x020، هذا العلم يشير ما إذا كانت وحدة التخزين تحتفظ بملفات محذوفة يمكن استعادتها (أنظر للحيد 0x0C في مدخلات الدليل)</td><td>1</td><td>36</td><td>0x026</td></tr>
<tr><td>الرقم التسلسلي للقرص في MSX-DOS 2 (القيمة الاعتيادية: 0x00000000). إذا كان التوقيع "VOL_ID" موجود عند حيد القطاع 0x020، نظام MSX-DOS 2 يخزن هنا الرقم التسلسلي للوحدة التخزين للكشف عن تغيير/تبديل الوسيط (القرص).</td><td>4</td><td>56 23 36 C0</td><td>0x027</td></tr>
<tr><td>محجوزة</td><td>5</td><td>00 00 00 00 00</td><td>0x02B</td></tr>
<tr><td>مدخلة شفرة MSX-DOS 2 من أجل معالجات Z80 داخل شفرة إقلاع MSX. هذا الموقع حيث تقفز أجهزة MSX-DOS 2 عند تمرير التحكم إلى قطاع الاقلاع. هذه الموقع يتداخل مع EBPB منذ DOS 4.0 / OS/2 1.2 أو شفرة قطاع الاقلاع المتوافقة مع x86 في قطاعات الاقلاع المتوافقة مع IBM PC، وسيؤدي إلى انهيار جهاز MSX ما لم تتخذ إجراءات وقائية خاصة كمثل وقف المعالج هنا في متكررة حلقية tight loop (شفرة التشغيل JR 0x030 for 0xFE 0x18).</td><td>متفاوت (2)</td><td></td><td>0x030</td></tr>
<tr><td>توقيع</td><td>2</td><td></td><td>0x1FE</td></tr>
</table>

تحليل جزء من بنية قطاع إقلاع MSX-DOS

تعليمة القفزة (3 بايت) تقفز إلى شفرة الاقلاع

لصيقة اسم وحدة التخزين (8 بايت)

منطقة معاملات DOS 3.0 BPB

عدد بايتات في كل قطاع (2 بايت)

عدد القطاعات في كل عنقود (1 بايت)

عدد القطاعات المنطقية المحجوزة (2 بايت)

عدد نسخ FAT في نظام الملفات (1 بايت)

عدد مدخلات الدليل الجذر (2 بايت)

عدد القطاعات الإجمالي على الوسيط (2 بايت)

واصف الوسيط (1 بايت)

عدد القطاعات في كل نسخة FAT (2 بايت)

عدد القطاعات الفيزيائية لكل مسار (2 بايت)

عدد الرؤوس على الوسيط (2 بايت)

عدد القطاعات المخفية (2 بايت)

منطقة الشفرة

شفرة إقلاع MSX-DOS 1، (متفاوت 2)

توقيع MSX-DOS 2 (حجم 6 بايت)

علم استعادة الملفات المحذوفة (1 بايت)

الرقم التسلسلي للقرص دوس MSX-DOS 2 (حجم 4 بايت)

محجوزة (5 بايت)

شفرة دوس MSX-DOS 2 (متفاوت 2)

رسائل الأخطاء وأسماء ملفات النظام

رسائل الأخطاء (37 بايت)

أسماء ملفات النظام (11 بايت)

توقيع (2 بايت)

```
     00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F  0123456789ABCDEF
0000 EB FE 90 53 41 4E 59 4F 32 2E 30 00 02 02 01 00 |...SANYO2.0.....|
0010 02 70 00 D0 02 F8 02 00 09 00 01 00 00 00 D0 ED |.p..............|
0020 53 59 C0 32 d0 C0 36 56 23 36 C0 31 1F F5 11 AB |SY.2..6V#6.1....|
0030 C0 0E 0F CD 7D F3 3C CA 63 C0 11 00 01 0E 1A CD |....}.<.c.......|
0040 7D F3 21 01 00 22 B9 C0 21 00 3F 11 AB C0 0E 27 |}.!.."..!.?....'|
0050 CD 7D F3 C3 00 01 58 C0 CD 00 00 79 E6 FE FE 02 |.}....X....y....|
0060 C2 6A C0 3A D0 C0 A7 CA 22 40 11 85 C0 CD 77 C0 |.j.:....."@....w.|
0070 0E 07 CD 7D F3 18 B4 1A B7 C8 D5 5F 0E 06 CD 7D |...}....... ...}|
0080 F3 D1 13 18 F2 42 6F 6F 74 20 65 72 72 6F 72 0D |.....Boot error.|
0090 0A 50 72 65 73 73 20 61 6E 79 20 6B 65 79 20 66 |.Press any key f|
00a0 6F 72 20 72 65 74 72 79 0D 0A 00 00 4D 53 58 44 |or retry....MSXD|
00b0 4F 53 20 20 53 59 53 00 00 00 00 00 00 00 00 00 |OS  SYS.........|
00c0 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
0000 eb fe 90 59 44 2d 36 34 30 20 20 00 02 02 01 00 |...YD-640 .....|
0010 02 70 00 d0 02 f8 02 00 09 00 01 00 00 00 18 10 |.p..............|
0020 56 4f 4c 5f 49 44 00 39 7d 03 63 00 00 00 00 00 |VOL_ID.9}.c.....|
0030 d0 ed 53 6a c0 32 72 c0 36 67 23 36 c0 31 1f f5 |..Sj.2r.6g#6.1..|
0040 11 ab c0 0e 0f cd 7d f3 3c 28 26 11 00 01 0e 1a |......}.<(&.....|
0050 cd 7d f3 21 01 00 22 b9 c0 21 00 3f 11 ab c0 0e |.}.!.."..!.?....|
0060 27 cd 7d f3 c3 00 01 69 c0 cd 00 00 79 e6 fe d6 |'.}....i....y...|
0070 02 f6 00 ca 22 40 11 85 c0 0e 09 cd 7d f3 0e 07 |...."@......}...|
0080 cd 7d f3 18 b8 42 6f 6f 74 20 65 72 72 6f 72 0d |.}...Boot error.|
0090 0a 50 72 65 73 73 20 61 6e 79 20 6b 65 79 20 66 |.Press any key f|
00a0 6f 72 20 72 65 74 72 79 0d 0a 24 00 4d 53 58 44 |or retry..$.MSXD|
00b0 4f 53 20 20 53 59 53 00 00 00 00 00 00 00 00 00 |OS  SYS.........|
00c0 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 |................|
```

مثال أخر من القرص الرئيسي (النسخة الانجليزية)

MSX-DOS 2.2 (MSX2 / 1988 / Ascii) MDOS22DE.ROM

نظام الملفات NTFS (نظام ملفات التقنية الجديدة New Technology File System)

نظام ملفات إن تي إف أس NTFS من مايكروسوفت، يستخدم رسميا ضمن عائلة أنظمة ويندوز إن تي، منذ ويندوز إن تي 3.1. نظام إن تي إف أس يتفوق على نظام جدول توزيع الملفات FAT ونظام HPFS في العديد من الميزات الجديدة مثل دعم ميتاداتا (البيانات الوصفية) واستخدام بنية بيانات متقدمة لتحسين الكفاءة، والوثوقية، وأداء قرص التخزين، بالإضافة إلى تحسين أمن الملفات باستخدام قائمة التحكم بالنفاذ ACL ونظام الملفات المزود بقيد حوادث JFS.

هذا النظام لم يصمم فقط لأداء العمليات الاعتيادية بشكل أسرع مثل القراءة والكتابة، بل حتى العمليات المتقدمة مثل استعادة نظام الملفات، على الأقراص الكبيرة.

لكن هذا النوع من أنظمة الملفات لا يستحسن استخدامه مع الأقسام التي لا تزيد مساحتها عن 400 ميغابايت لأنه يستخدم مقدار كبير من المساحة في هيكلية (تراكيب) النظام.

الجزء المركزي الأساسي لنظام الملفات NTFS هو جدول الملف الرئيسي MFT . نظام الملفات NTFS يقوم بحفظ عدة نسخ للأجزاء الحرجة والمهمة من MFT لحمايتها من الفساد أو الضياع كما يقوم باستخدام العناقيد في تخزين بيانات الملفات، وحجم العنقود هنا لا يتوقف على حجم القرص أو القسم حيث أن عنقود بحجم صغير 512 بايت يمكنه تمثيل حجم القرص أو القسم مهما كان حجمه ؛ 500 ميغابايت أو 5 جيجابايت كما أن استعمال حجم صغير للعناقيد لا يقلل فقط من المساحة المهدورة من القرص الثابت وإنما أيضا يقلل من عملية تجزئة الملفات ؛ لأن تجزئة الملف على عدة عناقيد غير متجاورة يسبب بطء في الوصول إلى ذلك الملف، ونظام NTFS يعطي أداء جيد مع الأقراص الكبيرة.

نظام الملفات NTFS يدعم التصليح الفوري للأخطاء Hot fixing ؛ ويستطيع آليا اكتشاف القطاعات الفاسدة وترميزها (بعلامة) بحيث لا تستخدم في المستقبل.

ومما سبق عرضه نرى أن نظام NTFS هو الأفضل للاستعمال مقارنة بالنظام FAT وذلك لتميزه بالصفات التالية :

1. استهلاك أقل للذاكرة مقارنة باستهلاك FAT.
2. فهرسة التصاميم أكثر فاعلية للملفات لكل دليل.
3. تشفير المعلومات الذي يحسن من قوة الأمان بشكل كبير.
4. التخزين البعيد والذي يوفر توسيعا لمساحة القرص عن طريق إمكانية الوصول إلى الوسائط القابلة للإزالة.
5. تسجيل الاسترداد لبيانات التعريف NTFS والذي يساعدك في استعادة المعلومات بسرعة عند حدوث فشل في الطاقة أو مشكلة في النظام.
6. الحصص النسبية للقرص والتي يمكن استخدامها لمراقبة مقدار مساحة القرص المستخدمة من قبل المستخدمين كأفراد والتحكم به.
7. التحجيم الأفضل للأقراص الكبيرة. إن الحد الأقصى لحجم قرص NTFS أكبر بكثير مقارنة بـ FAT وعند زيادة حجم القرص لا يؤدي ذلك إلى تخفيض الأداء كما يحدث مع FAT.
8. التحكم بالوصول إلى الملفات والمجلدات ودعم حسابات للمستخدمين محدودة, أما في FAT فكافة المستخدمين لهم حق الوصول لكافة الملفات بغض النظر عن نوع الحساب.
9. NTFS يعمل بشكل أفضل مع الأقراص الكبيرة ثم يليه في ذلك نظام ملفات FAT32.
10. إمكانية التحويل من FAT إلى NTFS أما إذا وقع العكس سيتم فقدان البيانات.

عند تهيئة وحدة التخزين بنظام ملفات NTFS سوف ينتج عن ذلك عدة ملفات نظام (ميتاداتا) مثل $LogFile، $Bitmap، $MFT (جدول الملف الرئيس)، وملفات أخرى، تتضمن معلومات عن كافة الملفات والمجلدات على وحدة التخزين NTFS. أول المعلومات على وحدة التخزين ستكون قطاع إقلاع القسم (ملف ميتاداتا $Boot)، الذي يبدأ عند القطاع 0 ويمكن أن يصل طوله إلى 16 قطاع. هذا الملف يصف المعلومات الأساسية لوحدة التخزين وموقع ملف ميتاداتا الرئيسي $MFT.

تخطيط وحدة التخزين NTFS بعد التهيئة

قطاع إقلاع	جدول الملفات الرئيسي	بيانات ملفات النظام	نسخة جدول الملفات الرئيسي

سجل إقلاع NTFS (قطاع الاقلاع + قطاعات شفرة الاقلاع)

في وحدات التخزين NTFS سجل الاقلاع بطول 7 قطاعات (منها 6 قطاعات لشفرة الاقلاع)، ورغم أن هناك 16 قطاع محجوزة من أجل سجل إقلاع القسم، القطاعات الثمانية الأخيرة شاغرة (تتضمن فقط بايت أصفار)، أنظمة ويندوز XP/2000 تخزن أيضا نسخة احتياطية من سجل إقلاع وحدة تخزين في القطاع الأخير في القسم ! [26].

سجل إقلاع مايكروسوفت NTFS، يوصف غالبا بأنه قطاع واحد، لكن في الواقع، هذا القطاع لا يستطيع إقلاع نظام مثل ويندوز أكس بي من دون القطاعات الستة الأخرى التي تشكل معظم شفرة الاقلاع في القسم. وثائق مايكروسوفت تقول أن نظام التشغيل يخصص أول 16 قطاع (تعرف بملف $Boot) من أجل قطاع الاقلاع و شفرة الاقلاع.

سجل الاقلاع FAT32		عنوان (خطي / فيزيائي)	محتوى [35]
قطاع الاقلاع	القطاع المنطقي (0) 1	CHS 0-1-1, LBA 63	القفزة، هوية صانعي القطع الأصلية، معاملات BPB/EBPB، شفرة الاقلاع، التوقيع
قطاعات شفرة الاقلاع	القطاع المنطقي (1) 2	CHS 0-1-2, LBA 64	شفرة إقلاع القطاعات من 3 (2) إلى 6 (5) لا شيء يميزها
	القطاع المنطقي (2) 3	CHS 0-1-3, LBA 65	
	القطاع المنطقي (3) 4	CHS 0-1-4, LBA 66	
	القطاع المنطقي (4) 5	CHS 0-1-5, LBA 67	
	القطاع المنطقي (5) 6	CHS 0-1-6, LBA 68	
	القطاع المنطقي (6) 7	CHS 0-1-7, LBA 69	هذا القطاع الأخير في شفرة الاقلاع ينتهي بـ 138 بايت كلها أصفار
قطاعات شاغرة (محجوزة)	القطاع المنطقي (7-15) 8 - 16	- - - - - - -	8 قطاعات غير مستخدمة

أول قطاع من 7 قطاعات مستخدمة فعليا يتضمن العناصر التالية

حقل	بايت	إزاحة
تعليمة القفزة	3	000h (0)
هوية صانعي القطع الأصلية OEM ID	8	003h (3)
معاملات كتلة BPB	25	00Bh (11)
معاملات الكتلة الممتدة EBPB	48	024h (36)
شفرة إقلاع ابتدائية	426	054h (84)
توقيع القطاع (علامة نهاية القطاع)	2	1FEh (510)

15 قطاع الأخرى (بالأحرى 6) المتبقية تتضمن فقط شفرة إقلاع إضافية، بدون التوقيع أو بنى أخرى. وباستثناء القطاع الأول جميع القطاعات ينبغي أن تكون نفسها على أي وحدة تخزين NTFS.

قطاعات "$Boot" في NTFS

هذا ملف بيانات وصفية (ميتاداتا) يشير إلى سجل إقلاع وحدة التخزين. أي أن 16 قطاع في قسم NTFS تعرف باسم "$Boot" هذا الملف يحتل العناقيد الأولى على وحدة التخزين ؛ العنقود 0 و العنقود 1 ؛ في معظم الأنظمة التي تستخدم 8 قطاعات للعنقود أو 4 كيلوبايت لكل عنقود (4096 بايت). ويتضمن معلومات عن وحدة التخزين في معاملات BPB مثل الحجم والرقم التسلسلي وعدد عناقيد ملف $MFT والملف المرآوي $MFTMirr، كما يتضمن شفرة للإقلاع مثل شفرة "NTLDR/BOOTMGR".

الكتلة الممتدة NTFS EBPB

في وحدات التخزين NTFS، بنية الكتلة الممتدة NTFS EBPB جديدة بالكامل. رغم أنها تتضمن بعض الحقول بنفس العناصر المستخدمة في الكتلة السابقة، لكن مايكروسوفت حذفت جميع الحقول السابقة بداية من الحيد 28h واستخدمت مكانها حقول أطول (بايت ثماني) مطلوبة في NTFS. بيانات هذه الحقول أثناء بدء التشغيل تخول برنامج محمل الاقلاع ntldr إيجاد جدول الملف الرئيسي MFT. في وحدات التخزين NTFS، غير أن الملف MFT لا يقع في قطاع معروف مسبقا، كما هو الحال مع وحدات التخزين FAT16 و FAT32. لهذا السبب، لا يمكن تحريك ملف MFT من مكانه إذا كان القطاع فاسد في الموقع المعتاد. على أية حال، إذا كانت البيانات فاسدة لا يمكن تحديد موقع ملف MFT، في هذه الحالة، نظام ويندوز سوف يفترض أن وحدة التخزين بدون تهيئة.

الجدول التالي يصف بنية أول قطاع إقلاع من 7 قطاعات تشكل طول سجل إقلاع القسم NTFS. (هذا الجدول من مايكروسوفت يختلف قليلا عن جدول الموسوعة)

إزاحة	رمز تذكري	بايت	مثال	وصف
000h (0)	BS_jmpBoot	3	EB 52 90	**تعليمة القفزة.** من أجل القفز إلى شفرة الاقلاع، (عادة تكون EB5290h وفق نظام الملفات وموقع شفرة الاقلاع). قيمة | وظيفة | بايت EB | تعليمة القفزة | قفزة قصيرة SHORT JMP | 1 52 | إزاحة نسبية | 1 90 | تعليمة لا عملية NOP. | 1
003h (3)	BS_OEMName	8	NTFS	**هوية صانعي القطع الأصلية** (OEM ID). يشير إلى النظام المستخدم في تهيئة وحدة التخزين. وهو NTFS. لكن النظام لا يستخدمها بعد التهيئة.
				بداية معاملات كتلة NTFS BPB
00Bh (11)	BPB_BytsPerSec	2	00 02	**عدد بايتات لكل قطاع.** حجم القطاع على الوسيط الفيزيائي، في العادة يكون 512 بايت. ويمكن أن يكون بإحدى القيم التالية : عشري | ست عشري | ترتيب بايت 512 | 200h | 00 02 1024 | 400h | 00 04 2048 | 800h | 00 08 4096 | 1000h | 00 10 للتوافق يجب أن تكون دائما 512 بايت، غالبا الأنظمة لن تحاول فحص هذه القيمة لأن معظم الشفرات مصممة لاستخدام 512 بايت في كل قطاع. ورغم أن أنظمة مايكروسوفت تدعم القيم الأخرى، لكن لا ينصح باستخدامها.
00Dh (13)	BPB_SecPerClus	1	08	**عدد القطاعات المنطقية لكل عنقود.** في ويندوز XP/2000 عادة تكون 8. والقيمة يجب أن يكون قوة العدد 2 أكبر من 0 ؛ بإحدى القيم التالية: عشري | ست عشري | بايت | كيلوبايت 1 | 1h | 512 | 0.5 2 | 2h | 1024 | 1 4 | 4h | 2048 | 2 8 | 8h | 4096 | 4 16 | 10h | 8192 | 8 32 | 20h | 16384 | 16 64 | 40h | 32768 | 32 128 | 80h | 65536 | 64
00Eh (14)	BPB_RsvdSecCnt	2	00 00	**عدد القطاعات المحجوزة** (يجب أن تكون دائما 0). عدد القطاعات المحجوزة قبل VBR. القيمة دائما تكون صفر لأن NTFS يضع قطاع الاقلاع في بداية القسم. إذا لم تكن 0، يفشل نظام ملفات NTFS في وصل وحدة تخزين. بعد تحميل في الذاكرة الموقع يصبح 7C0Eh وبخزن عدد قطاعات سجل الاقلاع التي ستقرأ؛ ويبدأ بالقيمة 16 (العشرية) ويحسب تنازليا إلى 0.
010h (16)	BPB_Reserved	1	00	**محجوزة** (دائما 0. أو سيفشل NTFS في وصل وحدة تخزين) هذا الحقل يستخدم من أجل عدد نسخ FAT في وحدة تخزين FAT12/16
011h (17)	BPB_Reserved	2	00 00	**محجوزة** (دائما 0. أو سيفشل NTFS في وصل وحدة تخزين) هذا الحقل يستخدم من أجل عدد مدخلات المجلد في الدليل الجذر في وحدة تخزين FAT12/16
013h (19)	BPB_Reserved	2	00 00	**محجوزة** (دائما 0. أو سيفشل NTFS في وصل وحدة تخزين) في مايكروسوفت، مشغل NTFS لا يدعم وحدات التخزين NTFS التي تتضمن 65535 قطاع أو أقل. لذلك يرفض وصل وحدات التخزين حين يتضمن هذا الحقل قيمة غير الصفر. هذا الحقل يستخدم من أجل عدد القطاعات الاجمالي (16-بت) في وحدة تخزين FAT12/16
015h (21)	BPB_Media	1	F8	**واصف الوسيط.** بايت واصف الوسيط؛ يوفر معلومات عن الوسيط المستخدم. عادة القيمة تكون F8h على كافة الوسائط الثابتة مثل القرص الثابت، وتكون F0h على معظم الوسائط التي تقبل الفصل مثل القرص المرن. مدخلات واصف الوسيط استخدمت في أقراص مايكروسوفت دوس FAT16 ولا تستخدم في أنظمة مثل خادوم ويندوز 2003 و XP/2000. لكن الشفرات الأخرى قد تستخدمها لهذا يتم تعيين هذا الحقل إلى القيمة F8.
016h (22)	BPB_Reserved	2	00 00	**محجوزة** (دائما 0. أو سيفشل NTFS في وصل وحدة تخزين) هذا الحقل يستخدم من أجل عدد القطاعات (16-بت) في كل FAT. في وحدة تخزين FAT12/16
018h (24)	BPB_SecPerTrk	2	3F 00	**عدد القطاعات لكل مسار.** وثائق مايكروسوفت الخاصة بنظام ملفات NTFS تقول أن هذا الحقل "غير مستخدم ولا يفحص من قبل NTFS". عدد القطاعات لكل مسار بقياسات القرص التي تستخدم نداء المقاطعة 13h. عادة تكون 63 على القرص الثابت.
01Ah (26)	BPB_NumHeads	2	FF 00	**عدد الرؤوس.** وثائق مايكروسوفت الخاصة بنظام ملفات NTFS تقول أن هذا الحقل "غير مستخدم ولا يفحص من قبل NTFS". عدد الرؤوس (الجوانب) التي تستخدم نداء المقاطعة 13h. عادة تكون 255 (FF) على القرص الثابت.

عدد القطاعات المخفية. القطاعات المخفية التي تسبق القسم الذي يتضمن وحدة التخزين. عادة تكون 63 (3Fh) لأول وحدة تخزين. • الحقل مرتبط فقط بالوسيط المرئي في المقاطعة 13h ويجب أن يكون 0 على الوسيط بدون أقسام. • رغم أن جميع تطبيقاتها في مايكروسوفت غير معروفة، هذه القيمة تستخدم أثناء قراءة قطاعات الاقلاع في الذاكرة ! • هذا القيمة من المفترض أن تكون عدد القطاعات الفيزيائية على القرص التي تسبق القطاع الأول لوحدة التخزين (أسطوانة 0 رأس 0)؛ هذا يفسر لماذا المدخلة الأولى في جدول أقسام القرص تملك القيمة 63، وستكون أيضا 63 لكل وحدة تخزين أولى في القسم الممتد لأنه يسبقها سجل إقلاع ممتد خاص. ولذلك القيمة تتفاوت فقط في NTFS إذا القسم الأولى هو الثاني، الثالث، الرابع وليس الأول.. • في ويندوز فيستا/7، عدد القطاعات المخفية أو المحجوزة لأول قسم ارتفع إلى 2048 (0x800) بدل 63. • القيمة 29 (1Dh) في هذا الحقل غالبا تشير إلى أن وحدة التخزين على قرص ديناميكي. وثائق مايكروسوفت الخاصة بنظام ملفات NTFS تقول أن هذا الحقل "غير مستخدم ولا يفحص من قبل NTFS". لكن يبدوا أن تلك الإفادة خاطئة. لأن شفرة إقلاع VBR في NTFS تستخدم هذا الحقل بنفس الأسلوب، الموصوف في نظام دوس 3.31. مع أنظمة FAT12، FAT16، FAT16B كما تفعل شفرة إقلاع VBR في جميع بنى أنظمة الملفات الأخرى وأنظمة التشغيل. التحويل بين العناوين المرتبطة بالقرص وتلك المرتبطة بوحدة التخزين هو عمل تقوم به جميع شفرات إقلاع VBR. ولا يمكن إلغاءه/تجنبه.	3F 00 00 00	4	BPB_HiddSec	01Ch (28)
محجوزة (دائما 0. أو سيفشل NTFS في وصل وحدة تخزين) يستخدم لعدد القطاعات الكبرى الإجمالي في وحدة تخزين FAT12/16، والعدد الإجمالي للقطاعات في وحدة تحزين FAT32	00 00 00 00	4	BPB_Reserved	020h (32)

نهاية معاملات كتلة NTFS BPB وبداية معاملات الكتلة الممتدة NTFS EBPB

محجوزة . وثائق مايكروسوفت الخاصة بنظام ملفات NTFS تقول أن هذا الحقل "غير مستخدم ولا يفحص من قبل NTFS. لكن نظام NTFS يعين دائما هذا الحقل إلى 80008000h. لحظ أن أول بايت 80h يمثل رقم القرص ! أيضا عند تفحص شفرة سجل الاقلاع (في الذاكرة)، ظهر أن أول بايت من هذه 4 بايت يستخدم لتعيين رقم القرص الفيزيائي.	80 00 80 00	4	Reserved	024h (36)
عدد القطاعات الإجمالي على وحدة التخزين. كما ذكرنا أعلاه، هذه القيمة دائما ستكون بحجم أقل بقطاع واحد من مجموع عدد القطاعات في مدخلة وحدة التخزين في جدول الأقسام، لأن قطاع النسخة الاحتياطية NTFS ليس جزء من وحدة التخزين NTFS. [26]	AE 39 D7 00 00 00 00 00	8	BPB_TotSec64	028h (40)
رقم العنقود المنطقي لملف $MFT. رقم العنقود المنطقي لبداية ملف $MFT في القسم. يحدد موقع ملف MFT باستخدام رقم العنقود المنطقي الخاص بالملف. (القطاع المنطقي 32 ؛ إذا كان عدد القطاعات في كل قمة عنقود هو 8).	04 00 00 00 00 00 00 00	8	BPB_MftClus	030h (48)
رقم العنقود المنطقي لملف $MFTMirr. رقم العنقود المنطقي لبداية ملف $MFTMirr في القسم. يحدد موقع النسخة المرآوية للملف MFT باستخدام رقم العنقود المنطقي للملف.	8D EF 00 00 00 00 00 00	8	BPB_MirClus	038h (56)
عدد العناقيد في كل سجل MFT. عدد العناقيد لكل سجل MFT. حجم كل سجل، (عدد العناقيد لكل قطعة سجل ملف FRS). NTFS ينشئ سجل ملف لكل ملف وسجل مجلد لكل مجلد يتم إنشاءه على وحدة تخزين NTFS. الملفات والمجلدات الأصغر من هذا الحجم تضمن في ملف MFT. • إذا كان هذا العدد إيجابي (قد يصل إلى 7Fh)، سوف يمثل عدد العناقيد لكل سجل MFT. • إذا كان العدد سلبي (من 80h إلى FFh)، حجم سجل الملف سيكون مضروب 2 في القيمة المطلقة لهذا العدد.	F6	1 عدد بإشارة (+/-)	BPB_ClusPerMft	040h (64)
محجوزة. عادة تكون 0. وثائق مايكروسوفت للنظام ملفات NTFS تقول أن هذا الحقل "لا يستخدم في NTFS.	00 00 00	3	Reserved	041h (65)
عدد العناقيد لكل صوان مفهرس. حجم كل صوان مفهرس، يستخدم لتخصيص حيز للأدلة (المجلدات). • إذا كان هذا العدد إيجابي (قد يصل حتى 7Fh)، سيمثل عدد العناقيد لكل سجل MFT. • إذا كان عدد سلبي (من 80h إلى FFh)، حجم سجل الملف سيكون مضروب 2 في القيمة المطلقة لهذا الرقم.	01	1	BPB_ClusPerIndx	044h (68)
محجوزة. عادة تكون 0. وثائق مايكروسوفت الخاصة بنظام ملفات NTFS تقول أن هذا الحقل "لا يستخدم في NTFS.	00 00 00	3	Reserved	045h (69)
الرقم التسلسلي لوحدة التخزين NTFS. لدعم تعقب وحدة التخزين على الوسيط القابل للإزالة. عادة تولد باستخدام التاريخ والوقت زمن التهيئة . عرض الرقم التسلسلي بالأمر DIR في CMD/DOS يعرض 4 بايت الأخيرة فقط من 8 بايت، مثال: C:\>dir Volume in drive C is myWindows Volume Serial Number is 827B-B23A	FC 5D E1 A4 99 E1 A4 B4	8	BS_VolID	048h (72)
محجوزة / تدقيق المجموع. وثائق مايكروسوفت الخاصة بنظام ملفات NTFS تقول أن هذا الحقل "غير مستخدم من قبل NTFS. هذا الحقل يتضمن فقط بايت أصفار لكن البعض يسمى هذا الحقل حقل تدقيق المجموع Checksum.	00 00 00 00	4	Reserved	050h (80)

نهاية معاملات الكتلة الممتدة NTFS EBPB وبداية شفرة الاقلاع وتوقيع الاقلاع

شفرة برنامج الاقلاع. أنظر أدناه إلى "عمل الشفرة في NTFS"	متفاوت	426	BS_BootCode	054h (84)
توقيع الاقلاع ؛ يجب أن تكون 55h AAh	55 AA	2	BS_Signature	1FEh (510)

وفق ألوان العمود الثاني في الجدول أعلاه:

بيانات هذه الحقول قد تتفاوت من حاسوب إلى آخر | بيانات هذه الحقول لا ينبغي أن تتفاوت بين أنظمة NTFS | بيانات هذه قد تكون مختلفة في بعض الظروف، لكن عادة تبقى هي نفسها كما تظهر في الجدول

الحقول التي تبدأ عند الحيود 0x0B, 0x0D, 0x15, 0x18, 0x1A, 0x1C تماثل الحقول الموجودة في وحدات تخزين FAT16 و FAT32. (أنظر للعمود الأخير أعلاه)

بنية القطاع الأول (في سجل إقلاع NTFS)

مثل بقية سجلات إقلاع أنظمة مايكروسوفت (منذ م.س-دوس 2)، 3 بايت الأولى تدعى تعليمة القفزة JMP. (مثال : EB 52 90) [28] في الواقع 2 بايت الأولى فقط هي تعليمة القفزة الفعلية إلى بقية شفرة التجميع التنفيذية في أنظمة x86 ؛ أما البايت الثالث 90h مجرد تعليمة لا عملية NOP.

بعد بايتات القفزة تأتي 8 بايت التالية تشكل هوية صانعي القطع الأصلية OEM ID أو اسم النظام ("NTFS" مع 4 فراغات) في وحدة تخزين NTFS ؛ متبوعة بمعاملات BPB.

رغم أن كتلة NTFS BPB تشترك مع سجلات إقلاع FAT16 و FAT32 في الكثير من الحقول ؛ مثل حقل عدد بايتات في كل قطاع، وعدد القطاعات لكل عنقود، ..إلخ. بل وحتى حقل بايت واصف الوسيط القديم (قيمة F8)، إلا أن، هذه الكتلة لا تتضمن حقول نوع النظام System ID أو لصيقة اسم وحدة التخزين Volume Label، ولا أية أسماء ملفات نظام. لكن، إذا راجعت الشفرة التي تتبع محمل الاقلاع الابتدائي ستلحظ عدد من ملفات النظام مثل BOOT.INI. في هذه الكتلة أيضا، استبدال رقم وحدة التخزين التسلسلي القديم 4-بايت برقم جديد 8 بايت.(أنظر للطرح أدناه).

بنية أول قطاع في سجل إقلاع NTFS (الجزء العلوي)

- تعليمة القفزة (اللامشروطة) (3 بايت)
- هوية صانعي القطع الأصلية (من أجل التعريف بالنظام المستخدم في التهيئة) (8 بايت)

منطقة معاملات BPB

- عدد بايتات في كل قطاع (2 بايت)
- واصف الوسيط (1 بايت)
- عدد القطاعات لكل عنقود (1 بايت)
- محجوزة، يجب أن تكون 0 (2 بايت)
- عدد القطاعات المحجوزة، يجب أن تكون 0 (2 بايت)
- لا يستخدمها NTFS ولا يتفحصها (2 بايت)
- محجوزة، يجب أن تكون 0 (1 بايت)
- لا يستخدمها NTFS ولا يتفحصها (2 بايت)
- محجوزة، يجب أن تكون 0 (2 بايت)
- لا يستخدمها NTFS ولا يتفحصها (4 بايت)
- محجوزة، يجب أن تكون 0 (2 بايت)
- محجوزة، يجب أن تكون 0 (4 بايت)

منطقة معاملات الممتدة EBPB

- محجوزة، يجب أن تكون 0 (4 بايت)
- محجوزة، يجب أن تكون 0 (3 بايت)
- عدد القطاعات الإجمالي على وحدة التخزين (8 بايت)
- عدد العناقيد لكل صوان فهرس (1 بايت)
- رقم عنقود بداية ملف $MFT في القسم (8 بايت)
- لا يستخدمها NTFS ولا يتفحصها، وتكون 0 (3 بايت)
- رقم عنقود بداية ملف $MFTMirr في القسم (8 بايت)
- الرقم التسلسلي لوحدة التخزين (8 بايت)
- عدد العناقيد لكل سجل مجلد/ملف MFT (1 بايت)
- لا يستخدمها NTFS ولا يتفحصها، وتكون 0 (4 بايت)

منطقة الشفرة

- شفرة إقلاع تنفيذية (426 بايت) سوف تكون مختلفة بين أنظمة التشغيل

(القطاع المطلق) CHS 0-1-**1**, LBA **63**

```
          00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F  0123456789ABCDEF
0000: EB 52 90 4E 54 46 53 20 20 20 20 00 02 08 00 00  [.R.NTFS    .....]
0010: 00 00 00 00 00 F8 00 00 3F 00 FF 00 3F 00 00 00  [........?...?...]
0020: 00 00 00 00 80 00 80 00 AE 39 D7 00 00 00 00 00  [.........9......]
0030: 04 00 00 00 00 00 00 00 8D EF 00 00 00 00 00 00  [................]
0040: F6 00 00 00 01 00 00 00 FC 5D E1 A4 99 E1 A4 E4  [.........]......]
0050: 00 00 00 00 FA 33 C0 8E D0 BC 00 7C FB B8 C0 07  [.....3.....|....]
0060: 8E D8 E8 16 00 B8 00 0D 8E C0 33 DB C6 06 0E 00  [..........3.....]
0070: 10 E8 53 00 68 00 0D 68 6A 02 CB 8A 16 24 00 B4  [..S.h..hj....$..]
0080: 08 CD 13 73 05 B9 FF FF 8A F1 66 0F B6 C6 40 66  [...s......f...@f]
0090: 0F B6 D1 80 E2 3F F7 E2 86 CD C0 ED 06 41 66 0F  [.....?.......Af.]
00A0: B7 C9 66 F7 E1 66 A3 20 00 C3 B4 41 BB AA 55 8A  [..f..f. ...A..U.]
00B0: 16 24 00 CD 13 72 0F 81 FB 55 AA 75 09 F6 C1 01  [.$...r...U.u....]
00C0: 74 04 FE 06 14 00 C3 66 60 1E 06 66 A1 10 00 66  [t......f`..f...f]
00D0: 03 06 1C 00 66 3B 06 20 00 0F 82 3A 00 1E 66 6A  [....f;. ...:..fj]
00E0: 00 66 50 06 53 66 68 10 00 01 00 80 3E 14 00 00  [.fP.Sfh.....>...]
00F0: 0F 85 0C 00 E8 B3 FF 80 3E 14 00 00 0F 84 61 00  [........>.....a.]
0100: B4 42 8A 16 24 00 16 1F 8B F4 CD 13 66 58 5B 07  [.B..$.......fX[.]
0110: 66 58 66 58 1F EB 2D 66 33 D2 66 0F B7 0E 18 00  [fXfX..-f3.f.....]
0120: 66 F7 F1 FE C2 8A CA 66 8B D0 66 C1 EA 10 F7 36  [f.......f..f....6]
0130: 1A 00 86 D6 8A 16 24 00 8A E8 C0 E4 06 0A CC B8  [......$.........]
0140: 01 02 CD 13 0F 82 19 00 8C C0 05 20 00 8E C0 66  [........... ...f]
0150: FF 06 10 00 FF 0E 0E 00 0F 85 6F FF 07 1F 66 61  [..........o...fa]
0160: C3 A0 F8 01 E8 09 00 A0 FB 01 E8 03 00 FB EB FE  [................]
0170: B4 01 8B F0 AC 3C 00 74 09 B4 0E BB 07 00 CD 10  [.....<.t........]
0180: EB F2 C3 0D 0A 41 20 64 69 73 6B 20 72 65 61 64  [.....A disk read]
0190: 20 65 72 72 6F 72 20 6F 63 63 75 72 72 65 64 00  [ error occurred.]
01A0: 0D 0A 4E 54 4C 44 52 20 69 73 20 6D 69 73 73 69  [..NTLDR is missi]
01B0: 6E 67 00 0D 0A 4E 54 4C 44 52 20 69 73 20 63 6F  [ng...NTLDR is co]
01C0: 6D 70 72 65 73 73 65 64 00 0D 0A 50 72 65 73 73  [mpressed...Press]
01D0: 20 43 74 72 6C 2B 41 6C 74 2B 44 65 6C 20 74 6F  [ Ctrl+Alt+Del to]
01E0: 20 72 65 73 74 61 72 74 0D 0A 00 00 00 00 00 00  [ restart.......]
01F0: 00 00 00 00 00 00 00 00 83 A0 B3 C9 00 00 55 AA  [..............U.]
```

القطاع الأول، وحدة التحزين NTFS، ويندوز XP/2000، كل سطر 16 بايت (32 سطر)

بنية أول قطاع في سجل إقلاع NTFS (الجزء السفلي)

رسائل الأخطاء والإزاحة

- رسائل الأخطاء (104 بايت !)
- إزاحة الرسالة (4 بايت)
- توقيع قطاع الاقلاع؛ دائما 55AAh (إذا كان القطاع سليم !)

125 بايت الأخيرة في القطاع الأول تتضمن رسائل الأخطاء، و بايتات إزاحة الرسائل وتوقيع الاقلاع AA55h [24]. 2 بايت 0Dh و 0Ah، تشير إلى بداية كل سطر رسالة (عبارة عن محرف رجوع إلى السطر وتزويد سطر، CRLF) تنتهي بـ 00h [27] مباشرة بعد قطاع إقلاع NTFS تأتي 6 قطاعات، تتضمن شفرة تتفاعل مع ملف ntldr (ملف يتبع محمل الاقلاع NTLDR في ويندوز أن تي)، هذه الشفرة تظل مطلوبة لإقلاع نظام ويندوز، حتى وإن كان ملف ntldr غير موجود في القسم الأصلي؛ لأن الشفرة ستبحث عن الملف في قسم أخر وتستخدمه.

قطاعات شفرة الاقلاع

في وحدات تخزين NTFS، القطاع الثاني (أي القطاع الأول في شفرة الاقلاع) يبدأ دائما بـ 16 بايت التالية:

تشكل في معظمها 5 محارف NTLDR باستخدام الترميز (يونيكود)، هنا "05 00" تعني أن سلسلة يونيكود التالية بطول 5 محارف. متبوعة بـ "04 00" ثم علامة الدولار $ بترميز يونيكود. السطر التالي يبدأ بـ"49 00 33 00 30 00" (في بعض سجلات ملفات ميتاداتا في NTFS هذه العشرة بايت تشكل سوية ملف ".$.I.3.0." في شفرة أسكي وهو عبارة عن مؤشر/فهرس من أجل أسماء الملفات، يستخدم من قبل الأدلة، منذ +3 NTFS).

05	00	4E	00	54	00	4C	00	44	00	52	00	04	00	24	00	49	00	33	00	30	00
.	.	N	.	T	.	L	.	D	.	R	.	.	.	$	.	I	.	3	.	0	.

تنفيذ الشفرة سوف يتحول من قطاع إقلاع وحدة التخزين NTFS التي تقبل الإقلاع إلى شفرة الاقلاع عند الحيد 6Ah (كما تظهر في الطرح أسفل)، في أول قطاع من شفرة إقلاع NTFS.

2 بايت 8C C8 سوف تشكل أول التعليمات في المعالج :mov ax,cs. ورغم أن الغرض الفعلي من كل بايت غير معروف في المنطقة الملونة بين 00h و 69h (أنظر للطرح)، المنطقة تستخدمها شفرة الاقلاع لتخزين الحسابات الوسيطة والبيانات عن القسم. عند قراءة أول مرة هذه الشفرة سوف يلفت انتباهك وجود تعليمة القفزة (EB 12 ؛ يتبعها 90 90) وتظن أن هناك قفزة في هذا القطاع عند الحيد 56h، لكن في الواقع، تنفيذ البرنامج في سجل إقلاع NTFS يقفز دائما مباشرة إلى هذا القطاع عند الحيد 6Ah (الذي يبدأ بالتعليمة mov ax,cs)؛ ولا توجد أية قفزة إلى الحيد 56h. والسؤال هو: لماذا وضعت 4 بايت تلك عند الحيد 56h تحديدا !؟.

أول قطاع (القطاع 2) من شفرة الاقلاع يشبه هذه البنية (الإزاحة تبدأ من بداية أول قطاع في شفرة الاقلاع)

CHS 0-1-**2**, LBA **64** (القطاع المطلق)

```
     00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F  0123456789ABCDEF
0000 05 00 4E 00 54 00 4C 00 44 00 52 00 04 00 24 00 [..N.T.L.D.R...$.]
0010 49 00 33 00 30 00 00 E0 00 00 00 30 00 00 00 00 [I.3.0.....0....]
0020 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [................]
0030 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [................]
0040 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [................]
0050 00 00 00 00 00 00 EB 12 90 90 00 00 00 00 00 00 [................]
0060 00 00 00 00 00 00 00 00 00 00 8C C8 8E D8 C1 E0 [................]
0070 04 FA 8B E0 FB E8 03 FE 66 0F B7 06 0B 00 66 0F [........f.....f.]
0080 B6 1E 0D 00 66 F7 E3 66 A3 4E 02 66 8B 0E 40 00 [....f..f.N.f..@.]
0090 80 F9 00 0F 8F 0E 00 F6 D9 66 B8 01 00 00 00 66 [.........f.....f]
00A0 D3 E0 EB 08 90 66 A1 4E 02 66 F7 E1 66 A3 52 02 [.....f.N.f..f.R.]
00B0 66 0F B7 1E 0B 00 66 33 D2 66 F7 F3 66 A3 56 02 [f.....f3.f..f.V.]
00C0 E8 71 04 66 8B 0E 4A 02 66 89 0E 22 02 66 03 0E [.q.f..J.f.."..f..]
00D0 52 02 66 89 0E 26 02 66 03 0E 52 02 66 89 0E 2A [R.f..&.f..R.f..*]
00E0 02 66 03 0E 52 02 66 89 0E 3A 02 66 03 0E 52 02 [.f..R.f..:.f..R.]
00F0 66 89 0E 42 02 66 B8 90 00 00 00 66 8B 0E 22 02 [f..B.f.....f.."."]
0100 E8 5F 09 66 0B C0 0F 84 57 FE 66 A3 2E 02 66 B8 [._.f....W.f...f.]
0110 A0 00 00 00 66 8B 0E 26 02 E8 46 09 66 A3 32 02 [....f..&..F.f.2.]
0120 66 B8 B0 00 00 00 66 8B 0E 2A 02 E8 34 09 66 A3 [f.....f..*..4.f.]
0130 36 02 66 A1 2E 02 66 0B C0 0F 84 24 FE 67 80 78 [6.f...f....$.g.x]
0140 08 00 0F 85 1B FE 67 66 8D 50 10 67 03 42 04 67 [......gf.P.g.B.g]
0150 66 0F B6 48 0C 66 89 0E 62 02 67 66 8B 48 08 66 [f..H.f..b.gf.H.f]
0160 89 0E 5E 02 66 A1 5E 02 66 0F B7 0E 0B 00 66 33 [..^.f.^.f.....f3]
0170 D2 66 F7 F1 66 A3 66 02 66 A1 42 02 66 03 06 5E [.f..f.f.f.B.f..^]
0180 02 66 A3 46 02 66 83 3E 32 02 00 0F 84 1D 00 66 [.f.F.f.>2......f]
0190 83 3E 36 02 00 0F 84 C8 FD 66 8B 1E 36 02 1E 07 [.>6......f..6...]
01A0 66 8B 3E 46 02 66 A1 2A 02 E8 BC 01 66 0F B7 0E [f.>F.f.*....f...]
01B0 00 02 66 B8 02 02 00 00 E8 FE 07 66 0B C0 0F 84 [..f........f....]
01C0 A8 09 67 66 8B 00 1E 07 66 8B 3E 3A 02 E8 31 06 [..gf....f.>:..1.]
01D0 66 A1 3A 02 66 BB 20 00 00 00 66 B9 00 00 00 00 [f.:.f. ...f.....]
01E0 66 BA 00 00 00 00 E8 D6 00 66 85 C0 0F 85 23 00 [f........f....#.]
01F0 66 A1 3A 02 66 BB 80 00 00 00 66 B9 00 00 00 00 [f.:.f.....f.....]
```

وحدة التحزين NTFS، ويندوز XP/2000. القطاع الثاني (أي القطاع الأول في شفرة الاقلاع)

القطاعات من 3 إلى 6 لا شيء يميزها، أما القطاع الأخير 7 فينتهي بـ 138 بايت كلها أصفار (300 بايت في الشفرة الأصلية، قبل حزمة ويندوز أكس بي SP2). (أنظر للطرح).

في وحدة التخزين NTFS المهيئة حديثا، جدول الملف الرئيسي $MFT [29] يأتي مباشرة بعد منطقة $Boot، الجدول عادة يتضمن عدد من القطاعات محشوة بالبايت FFh (غالبا جزء من $Bitmap) وقبل أن تجد شيء يمكنك قراءته ! معظم وحدات التخزين NTFS سوف تتضمن أيضا في وسط القسم بعض البيانات الوصفية/ميتاداتا، (ملفات للنظام)؛ كمثل، $MFTMirr (وهي نسخة احتياطية من تسجيلات $MFT الأربعة الأولى) وملف $LogFile (الذي هو قيد حوادث NTFS ويمكن أن يكون بحجم عدة ميغابايت).

القطاع الأخير (القطاع السادس في شفرة الاقلاع ؛ أي القطاع السابع في كامل سجل إقلاع NTFS) سوف يبدو كالتالي:

القطاع 6 الأخير (القطاع 7) في قطاعات شفرة الاقلاع، وحدة التحزين NTFS، ويندوز XP/2000.

CHS 0-1-**7**, LBA **69** (القطاع المطلق)

```
     00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F  0123456789ABCDEF
0C00 10 67 66 8B 42 18 66 33 D2 66 F7 36 5E 02 66 33 [.gf.B.f3.f.6^.f3]
0C10 F6 66 50 66 56 66 58 66 5E 66 3B C6 0F 84 3A 00 [.fPfVfXf^f;...:.]
0C20 66 56 66 40 66 50 66 48 E8 1B FE 72 E8 E8 EB FD [fVf@fPfH...r....]
0C30 66 5A 66 5E 66 59 66 5B 66 53 66 51 66 56 66 52 [fZf^fYf[fSfQfVfR]
0C40 66 A1 42 02 67 66 8D 40 18 E8 D0 F8 66 0B C0 74 [f.B.gf.@....f..t]
0C50 C4 66 59 66 59 66 59 66 59 C3 66 59 66 59 66 33 [.fYfYfYfY.fYfYf3]
0C60 C0 C3 66 51 66 50 66 B8 05 00 00 00 1E 07 66 8B [..fQfPf.......f.]
0C70 F9 E8 8D FD 66 8B C1 66 BB 20 00 00 00 66 B9 00 [....f..f. ...f..]
0C80 00 00 00 66 BA 00 00 00 00 E8 33 F8 66 5B 66 59 [...f......3.f[fY]
0C90 66 85 C0 0F 85 15 00 66 8B C1 66 0F B7 0E 0C 02 [f......f..f.....]
0CA0 66 BA 0E 02 00 00 E8 16 F8 EB 33 90 66 33 D2 66 [f.........3.f3.f]
0CB0 8B C1 66 8B CB 66 50 66 53 E8 23 00 66 5B 66 5F [..f..fPfS.#.f[f_]
0CC0 66 0B C0 0F 84 17 00 1E 07 E8 35 FD 66 8B C7 66 [f.........5.f..f]
0CD0 0F B7 0E 0C 02 66 BA 0E 02 00 00 E8 E1 F7 C3 66 [.....f.........f]
0CE0 52 66 51 66 BB 20 00 00 00 66 B9 00 00 00 00 66 [RfQf. ...f.....f]
0CF0 BA 00 00 00 00 E8 C7 F7 66 0B C0 0F 84 63 00 66 [........f....c.f]
0D00 8B D8 1E 07 66 8B 3E 16 02 66 33 C0 E8 59 F8 1E [....f.>..f3..Y..]
0D10 07 66 8B 1E 16 02 66 59 66 5A 26 66 39 0F 0F 85 [.f....fYfZ&f9...]
0D20 0C 00 26 66 39 57 08 0F 84 31 00 EB 13 90 26 66 [..&f9W...1....&f]
0D30 83 3F FF 0F 84 2F 00 26 83 7F 04 00 0F 84 26 00 [.?.../.&......&.]
0D40 26 66 0F B7 47 04 03 D8 8B C3 25 00 80 74 CB 8C [&f..G.....%..t..]
0D50 C0 05 00 08 8E C0 81 E3 FF 7F EB BE 26 66 8B 47 [............&f.G]
0D60 10 C3 66 59 66 5A 66 33 C0 C3 A0 F9 01 E9 F4 F3 [..fYfZf3........]
0D70 A0 FA 01 E9 EE F3 00 00 00 00 00 00 00 00 00 00 [................]
0D80 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [................]
*
0DF0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [................]
```

(الإزاحة من بداية أول قطاع في شفرة الاقلاع، و بايت الأخير في الشفرة سيكون عند الحيد **0D75h**)

عمل الشفرة في NTFS

شفرة سجل الاقلاع الرئيسي تحمل سجل إقلاع NTFS [30] في موقع الذاكرة 0000:7C00. الكتلة NTFS BPB سوف تحتل عناوين الذاكرة من 7C0Bh إلى 7C53h، والشفرة التنفيذية لهذا القطاع ستكون في 303 بايت التالية من العنوان 7C54h إلى 7D82h وتتضمن البرنامج الرئيسي والعديد من الروتينات الثانوية التي بدورها سوف تحمل قطاعات إقلاع N.T.L.D.R في الذاكرة.

(انتبه! هذه القطاعات وليس ملف NTLDR الموجود في الدليل الجذر في القسم، أنظمة ويندوز أن تي).

نظرا لأن عمل النظام يعتمد عادة على قطاع الاقلاع للوصول إلى وحدة التخزين، يوصى بفحص وحدة التخزين بانتظام باستخدام برنامج مثل Chkdsk وعمل نسخ احتياطي للبيانات والملفات الخاصة لحمايتها من الضياع في حالة تعذر الوصول إلى القسم. لمعلومات أكثر راجع دليل استخدام Chkdsk.

قطاع الإقلاع في لينكس (كتلة التحكم في الاقلاع)

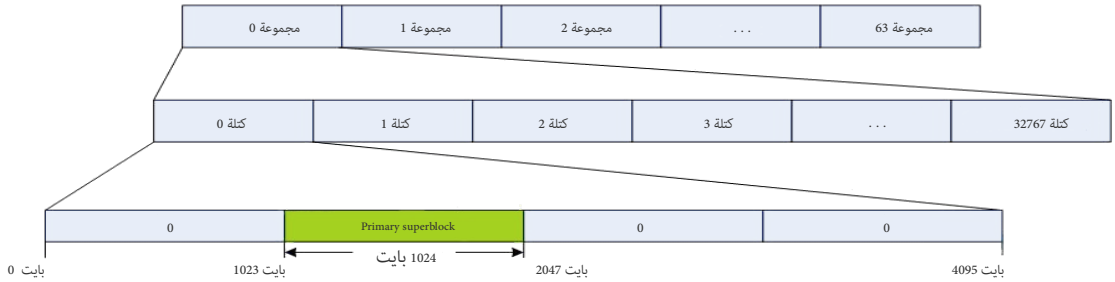
تقليديا توزيعات لينكس لا تملك شفرة إقلاع BCB في نظام الملفات (وحدة التخزين) إلى جانب النواة. وتستخدم عوض ذلك شفرة الاقلاع الابتدائية في قطاع MBR. (راجع كتيب MBR و GPT)،

رغم ذلك أول قطاعين (1.024 بايت) من بايت 0 إلى بايت 1023 قبل بداية كتلة نظام الملفات العليا super block، ستكون محجوزة من أجل شفرة قطاع الإقلاع x86، لكنها لا تستخدم، وقد تتضمن بيانات مخفية!.

حجم القطاع	أقصى سعة	نوع التنصيب	إقلاع	
512×2^{32} بايت	2.2 تيرابايت	BIOS-MBR	شفرة قطاع الإقلاع 0 + قطاعات بعد MBR (عادة تكون على الأقل 31 كيلوبايت أي 62 قطاع)	
		UEFI-MBR		
512×2^{64} بايت		BIOS-GPT	BBP + GRUB 2	شفرة الإقلاع في PMBR / Hybrid MBR
	9.4 زيتابايت	UEFI-GPT	مدير الإقلاع + ESP	

الكتلة العليا Super block

هذه كتلة من أجل التحكم في وحدة التخزين VCB (هذه الكتلة موجودة أيضا في أنظمة مثل مينكس و UFS وتشبه أيضا MFT في NTFS)، وتتضمن معلومات ضرورية لإقلاع نظام لينكس. لذلك توجد منها عدة نسخ احتياطية.لكن النسخة الأول فقط في أول كتلة من Block Group 0 يتم قراءتها عند وصل نظام الملفات (وحدة التخزين)، وتستخدم في الإقلاع. معلومات هذه الكتلة تسمح للمدير استخدام وصيانة النظام. نسخ من superblock و group descriptors ستكون فقط في المجموعة 0 و 1 وأس العدد 3، 5، 7، 9، 25، 27 إلى آخره. لكن في حالة تعطيل ميزة sparse_super، النسخ المكررة ستكون في جميع مجموعات الكتل. بينما تمكين الميزة sparse_super2 يسمح بوجود نسختين فقط من superblock و group descriptors. عادة تكون إحداها في بداية المجموعة #1 block group، والأخرى في المجموعة الأخيرة في نظام الملفات. هذه الميزة الأخيرة تسمح بزيادة نسبة كتل البيانات المتماسة على القرص! للملفات. (راجع أيضا ميزة flex_bg)، (لفهم وظيفة هذه المصطلحات، راجع كتيب ext4).



مثال : موقع الكتلة العليا super block على نظام ملفات يملك 63 مجموعة (هنا حجم الكتلة كان 4096 بايت) كل مجموعة بحجم 32768 كتلة باستثناء المجموعة الأخيرة

كتلة superblock تقع دائما عند بايت 1024 من بداية وحدة التخزين ودائما تكون بحجم ثابت 1024 بايت (مهما كان حجم الكتلة) في حالة قطاع 512 بايت، تبدأ عند الكتلة 2 LBA وتشغل القطاعات 2 و 3 :

بنية الكتلة العليا Super block ستكون struct ext4_super_block (من بايت 1024 إلى بايت 2047 في بداية مجموعة الكتل 0)

```
dd if=/dev/sda1 bs=1024 count=1 skip=1 | hexdump -Cv
        00 01 02 03 04 05 06 07  08 09 0A 0B 0C 0D 0E 0F  0123456789ABCDEFF

0000  00 00 38 00 00 96 df 00  e6 2d 0b 00 22 68 48 00  |..8.....-.."hH.|
0010  41 3a 33 00 00 00 00 00  02 00 00 00 02 00 00 00  |A:3.............|
0020  00 80 00 00 00 80 00 00  00 20 00 00 2a 73 ba 5a  |......... ..*s.Z|
0030  09 7b be 5a 02 00 04 00  53 ef 01 00 01 00 00 00  |.{.Z....S.......|
0040  79 5e ba 5a 00 00 00 00  00 00 00 00 01 00 00 00  |y^.Z............|
0050  00 00 00 00 0b 00 00 00  00 01 00 00 3c 00 00 00  |............<...|
0060  42 02 00 00 7b 00 00 00  91 2b 02 40 5b 79 47 c2  |B...{....+.@[yG.|
0070  a4 8b 75 a6 e6 d5 cf f3  00 00 00 00 00 00 00 00  |..u.............|
0080  00 00 00 00 00 00 00 00  2f 00 61 72 67 65 74 00  |......../.arget.|
0490  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
00a0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
00b0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
00c0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 fc 03  |................|
00d0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
00e0  08 00 00 00 00 00 00 00  00 00 00 00 96 63 e8 05  |.............c..|
00f0  1f d6 4f 28 8c 0d 03 33  f7 62 63 97 01 01 00 00  |..O(...3.bc.....|
0100  0c 00 00 00 00 00 00 00  d0 9d 37 58 0a f3 02 00  |..........7X....|
0110  04 00 00 00 00 00 00 00  00 00 00 00 ff 7f 00 00  |................|
0120  00 80 68 00 ff 7f 00 00  01 00 00 00 ff ff 68 00  |..h...........h.|
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 08  |................|
0150  00 00 00 00 00 00 00 00  00 00 00 00 1c 00 1c 00  |................|
0160  01 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
0170  00 00 00 00 04 00 00 00  86 4e d9 0a 00 00 00 00  |.........N......|
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 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  |................|
0200  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
0210  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
0220  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
0230  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
0240  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
0250  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
0260  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
0270  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
[Removed]
03f0  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
```

	حجم / نوع		إزاحة	رمز تذكري
عدد inodes الإجمالي في نظام الملفات	4	__le32	0x00 (00)	s_inodes_count
عدد الكتل الإجمالي في نظام الملفات	4	__le32	0x04 (04)	s_blocks_count_lo
عدد الكتل المحجوزة لمنع شغل نظام الملفات (32 بت المنخفضة) (سيكون من تخصيص المستخدم الجذر؛ ذو الصلاحيات العليا) (أنظر للحيد 0x50)	4	__le32	0x08 (08)	s_r_blocks_count_lo
عدد الكتل الحرة (الغير مخصصة)	4	__le32	0x0C (12)	s_free_blocks_count_lo
عدد inodes الحرة (الغير مخصصة)	4	__le32	0x10 (16)	s_free_inodes_count
أول كتلة بيانات (تتضمن Super Block) (ستكون 0 بالنسبة لجميع أحجام الكتلة، باستثناء كتلة 1 كيلوبايت التي يجب أن تكون 1 على الأقل)	4	__le32	0x14 (20)	s_first_data_block
حجم الكتلة: 2 ^ (10 + s_log_block_size)	4	__le32	0x18 (24)	s_log_block_size
حجم العنقود بعدد الكتل (2 ^ s_log_cluster_size). هذا في حالة تمكين ميزة bigalloc، ما عدا ذلك. s_log_cluster_size يجب أن يساوي s_log_block_size	4	__le32	0x1C (28)	s_log_cluster_size
عدد الكتل لكل مجموعة	4	__le32	0x20 (32)	s_blocks_per_group
عدد العناقيد لكل مجموعة، في حالة تمكين ميزة bigalloc. ما عدا ذلك. s_clusters_per_group يجب أن يساوي s_blocks_per_group	4	__le32	0x24 (36)	s_clusters_per_group
عدد inodes لكل مجموعة	4	__le32	0x28 (40)	s_inodes_per_group
زمن وصل نظام الملفات (وحدة التخزين) آخر مرة بعدد الثواني (توقيت يونكس)	4	__le32	0x2C (44)	s_mtime
زمن الكتابة إلى نظام الملفات آخر مرة بعدد الثواني (توقيت يونكس)	4	__le32	0x30 (48)	s_wtime
عدد مرات وصل نظام الملفات (وحدة التخزين) منذ أخر فحص fsck (منذ استخدام أداة فحص تماسك / ثبات نظام الملفات!)	2	__le16	0x34 (52)	s_mnt_count
عدد مرات وصل نظام الملفات المطلوبة قبل فحص fsck. (فحص تماسك نظام الملفات)	2	__le16	0x36 (54)	s_max_mnt_count
توقيع سحري! (لتأكيد وجود نظام الملفات EXT2/3/4 على وحدة التخزين): 0xEF53 — EXT2_SUPER_MAGIC	2	__le16	0x38 (56)	s_magic
(أعلام) حالة نظام الملفات القيم الصالحة ستكون:				s_state
نظام الملفات نظيف (مفصول على نحو نظيف): 0x0001 — EXT4_VALID_FS — Unmounted cleanly				
أخطاء في نظام الملفات: 0x0002 — EXT4_ERROR_FS — Errors detected				
استعادة inodes يتيمة: 0x0004 — EXT4_ORPHAN_FS — Orphans being recovered				
طريقة معالجة الخطأ ستكون إحدى القيم الثلاثة التالية:	2	__le16	0x3C (60)	s_errors
الاستمرار (تجاهل الخطأ): 1 — EXT4_ERRORS_CONTINUE — Continue execution				
إعادة وصل نظام الملفات في وضعية القراءة فقط: 2 — EXT4_ERRORS_RO — Remount fs read-only				
خطأ فادح داخلي (نواة النظام في وضعية Panic): 3 — EXT4_ERRORS_PANIC — Panic				
مستوى مراجعة ثانوي (قيمة 16 بت) داخل مستوى المراجعة (راجع أيضا: نظام التحكم بالمراجعات)	2	__le16	0x3E (62)	s_minor_rev_level
زمن الفحص الأخير، بعدد الثواني (توقيت يونكس)	4	__le32	0x40 (64)	s_lastcheck
الفترة الأقصى بين الفحوص، بعدد الثواني	4	__le32	0x44 (68)	s_checkinterval
نظام التشغيل / هوية نظام التشغيل الذي عن طريقه تم إنشاء نظام الملفات على وحدة التخزين .				s_creator_os
لينكس: 0 — EXT4_OS_LINUX — Linux				
حنو هيرد: 1 — EXT4_OS_HURD — Hurd				
اسم نظام تشغيل من تطوير Rémy Card: 2 — EXT4_OS_MASIX — Masix				
فري بي اس دي FreeBSD: 3 — EXT4_OS_FREEBSD — FreeBSD				
النظم المبنية على الإصدار BSD4.4-Lite: 4 — EXT4_OS_LITES — Lites				
مستوى المراجعة. ستكون إحدى هذه:	4	__le32	0x4C (76)	s_rev_level
صيغة أصلية: 0 — EXT4_GOOD_OLD_REV — The good old (original) format				
صيغة 2 مع أحجام inode ديناميكية وخصائص ممتدة....إلى آخره: 1 — EXT4_DYNAMIC_REV — V2 format w/ dynamic inode sizes				
معرّف المستخدم، الذي يستطيع استخدام الكتل المحجوزة: 0 — EXT4_DEF_RESUID — نوع المعرّف : UID يشير إلى: المستخدم الجذري	2	__le16	0x50 (80)	s_def_resuid
معرّف المجموعة، التي تستطيع استخدام الكتل المحجوزة: 0 — EXT4_DEF_RESGID — نوع المعرّف : GID يشير إلى: المجموعة الجذرية	2	__le16	0x52 (82)	s_def_resgid
الحقول التالية فقط من أجل superblocks EXT4_DYNAMIC_REV الاختلاف بين مجموعة الميزات المتوافقة compatible feature set والغير متوافقة incompatible feature set سيكون كالتالي: في حالة تعيين بت مجهول للنواة في الميزات الغير متوافقة، النواة يجب أن ترفض وصل نظام الملفات.. بينما في E2fsck يتم الغاء الميزة إذا كانت مجهولة للأداة سواء في الميزات المتوافقة أو الغير متوافقة				
أول inode غير محجوز في نظام الملفات	4	__le32	0x54 (84)	s_first_ino
حجم بنية inode بعدد بايتات (راجع فقرة: Inode Size)	2	__le16	0x58 (88)	s_inode_size
رقم مجموعة الكتل التي تنتمي إليها هذه الكتلة العليا superblock	2	__le16	0x5A (90)	s_block_group_nr
مجموعة أعلام الميزّات المتوافقة: Compatible feature set flags :	4	__le32	0x5C (92)	s_feature_compat
كتل الدليل المخصصة مسبقا من أجل خفض التجزئة (عند إنشاء دليل جديد) (أنظر للحيد 0xCD): 0x0001 — COMPAT_DIR_PREALLOC				

				0x0002	COMPAT_IMAGIC_INODES	"imagic inodes" ليس واضح ماذا يفعل هذا العلم (لكنه يشير إلى وجود inodes خادوم AFS)
				0x0004	COMPAT_HAS_JOURNAL	نظام ملفات مزود بقيد حوادث (سجل دوري)
				0x0008	COMPAT_EXT_ATTR	دعم الخصائص الممتدة (inodes تملك خصائص ممتدة)
				0x0010	COMPAT_RESIZE_INODE	يملك كتل GDT من أجل توسع نظام الملفات (نظام الملفات يستطيع إعادة تحجيم نفسه إلى أقسام أكبر)
				0x0020	COMPAT_DIR_INDEX	يملك فهارس للدليل (تستخدم شجرة b-trees hashed لتسريع عمليات البحث في الأدلة الكبيرة)
				0x0040	COMPAT_LAZY_BG	"Lazy BG" هذه الميزة تبدوا من أجل مجموعات الكتل الغير مهيئة؟ uninitialized block groups
				0x0080	COMPAT_EXCLUDE_INODE	"Exclude inode" (غير مستخدمة)
				0x0100	COMPAT_EXCLUDE_BITMAP	للإشارة إلى وجود exclude bitmaps ذات الصلة بصور snapshot ـ(غير مستخدمة في النواة / e2fsprogs)
				0x0200	COMPAT_SPARSE_SUPER2	إذا تم تعيين هذا العلم، حقل s_backup_bgs سيشير إلى مجموعتان فقط من الكتل تتضمنان نسخ superblock
s_feature_incompat	0x60 (96)	__le32	4	Incompatible feature set :		مجموعة أعلام الميزّات الغير متوافقة
				0x0001	INCOMPAT_COMPRESSION	ضغط البيانات
				0x0002	INCOMPAT_FILETYPE	مدخلات الدليل تتضمن حقل نوع الملف (راجع ext4_dir_entry_2)
				0x0004	INCOMPAT_RECOVER	نظام الملفات يحتاج إلى استعادة Filesystem needs recovery
				0x0008	INCOMPAT_JOURNAL_DEV	نظام الملفات يملك جهاز قيد حوادث منفصل
				0x0010	INCOMPAT_META_BG	مجموعات الكتل الوصفية (راجع الميزة: Meta Block Groups)
				0x0040	INCOMPAT_EXTENTS	الملفات تستخدم مديات في نظام الملفات (دعم المديات extents)
				0x0080	INCOMPAT_64BIT	تمكين حجم نظام الملفات 2^{64} كتلة
				0x0100	INCOMPAT_MMP	حماية نظام الملفات من الوصل المتعدد MMP. (غير مطبق) (راجع فقرة MMP)
				0x0200	INCOMPAT_FLEX_BG	مجموعات الكتل المرنة. (راجع الميزة: Flexible Block Groups)
				0x0400	INCOMPAT_EA_INODE	inodes يمكن استخدامها في تخزين قيم الخصائص الممتدة الكبيرة (راجع الميزة EA INODE)
				0x1000	INCOMPAT_DIRDATA	بيانات في مدخلة الدليل dirent (غير مطبق؟)
				0x2000	INCOMPAT_CSUM_SEED	بذرة تدقيق مجموع البيانات الوصفية مخزنة في الكتلة العليا superblock
				0x4000	INCOMPAT_LARGEDIR	دليل كبير > 2 جيجابايت أو مستوى 3 في شجرة Htree
				0x8000	INCOMPAT_INLINE_DATA	بيانات في inode
				0x10000	INCOMPAT_ENCRYPT	وجود inodes مشفرة على نظام الملفات
s_feature_ro_compat	0x64 (100)	__le32	4	Readonly-compatible feature set :		مجموعة أعلام الميزّات المتوافقة— في وضعية القراءة فقط
				0x0001	RO_COMPAT_SPARSE_SUPER	توصيف المجموعات ونسخ الكتلة العلياستكون متناثرة Sparse superblocks (أي ليست في كل المجموعات)
				0x0002	RO_COMPAT_LARGE_FILE	نظام الملفات يستخدم في تخزين ملفات أكبر من 2 جيجابايت
				0x0004	RO_COMPAT_BTREE_DIR	محتوى الدليل مخزن في شكل شجرة ثنائية أو BTREE !؟ (غير مستخدمة في النواة أو حزمة e2fsprogs)
				0x0008	RO_COMPAT_HUGE_FILE	النظام يملك أحجام ملفات تمثل بوحدات من الكتل المنطقية، وليس قطاع 512 بايت، هذا يدل عليه الملف الكبير جدا.
				0x0010	RO_COMPAT_GDT_CSUM	توصيف المجموعات Group descriptors يملك تدقيق مجامع
				0x0020	RO_COMPAT_DIR_NLINK	حد الأدلة الثانوية 32,000 في ext3 لم يعد مطبق. و i_links_count في الدليل يعين إلى 1 إذا زاد عن 64,999
				0x0040	RO_COMPAT_EXTRA_ISIZE	تشير إلى وجود inodes كبيرة على نظام الملفات
				0x0080	RO_COMPAT_HAS_SNAPSHOT	نظام الملفات يملك صورة snapshot
				0x0100	RO_COMPAT_QUOTA	تمكين نظام الحصة (الحصص النسبية للقرص)QUOTA
				0x0200	RO_COMPAT_BIGALLOC	نظام الملفات يدعم bigalloc، هذا يعني تعقب مديات الملف باستخدام وحدات من العناقيد (من الكتل) بدل الكتل
				0x0400	RO_COMPAT_METADATA_CSUM	دعم تدقيق مجموع البيانات الوصفية.(يقتضي ضمنا GDT_CSUM مع ذلك لا يجب تعيين GDT_CSUM)
				0x0800	RO_COMPAT_REPLICA	نظام الملفات يدعم النسخ طبق الأصل (هذه الميزة ليست في النواة ولا في e2fsprogs.)
				0x1000	RO_COMPAT_READONLY	صورة نظام ملفات للقراءة فقط؛ النواة لن تصلها في وضعية القراءة والكتابة ومعظم الأدوات لن تكتب إلى الصورة
				0x2000	RO_COMPAT_PROJECT	نظام الملفات يتعقب حصص القرص باستخدام project quotas (هذا نوع جديد من الحصص !)
s_uuid[16]	0x68 (104)	__u8	16	معرّف وحدة التخزين (بقيمة 128-بت UUID) (كما يظهر في خَرْج blkid ويجب أن يكون فريد)		
s_volume_name[16]	0x78 (120)	char	16	اسم وحدة التخزين Volume label (قيمة 16 بايت، ترميز أسكي / ISO-Latin-1 ينتهي بـ 0) (غير مستخدم تقريبا !)		
s_last_mounted[64]	0x88 (136)	char	64	مسار آخر نقط وصل، أي الدليل أين تم وصل نظام الملفات آخر مرة (هذه قيمة 64 بايت، ترميز أسكي / ISO-Latin-1 تنتهي بـ 0 للتوافق)		
s_algorithm_usage_bitmap	0xC8 (200)	__le32	4	خوارزمية ضغط البيانات. قيمة 32 بت من أجل تحديد طريقة للضغط البيانات (غير مستخدم في e2fsprogs / لينكس)		
				0	EXT2_LZV1_ALG	0x00000001
				1	EXT2_LZRW3A_ALG	0x00000002
				2	EXT2_GZIP_ALG	0x00000004

				3	EXT2_BZIP2_ALG	0x00000008	
				4	EXT2_LZO_ALG	0x00000010	

تنويه: التخصيص المسبق للدليل ينبغي أن يحدث فقط في حالة تمكين علم EXT4_FEATURE_COMPAT_DIR_PREALLOC.

s_prealloc_blocks	0xCC (204)	__u8	1	التخصيص المسبق للكتل	عدد الكتل المخصص مسبقا عند إنشاء ملفات اعتيادية (قيمة 8 بت) (غير مستخدم في e2fsprogs / لينكس)
s_prealloc_dir_blocks	0xCD (205)	__u8	1		عدد الكتل المخصص مسبقا للأدلة (قيمة 8 بت) (غير مستخدم في e2fsprogs / لينكس)
s_reserved_gdt_blocks	0xCE (206)	__le16	2		عدد المدخلات المحجوزة GDT من أجل توسع نظام الملفات مستقبلا

دعم نظام ملفات قيد الحوادث سيكون صالح في حالة تعيين EXT4_FEATURE_COMPAT_HAS_JOURNAL

s_journal_uuid[16]	0xD0 (208)	__u8	16	نظام ملفات قيد الحوادث (jbd2)	معرّف كتلة journal superblock التي تقع بعد superblock في قيد الحوادث الخارجي (قيمة 16 بايت UUID)
s_journal_inum	0xE0 (224)	__le32	4		رقم inode للملف قيد الحوادث journal file (قيمة 32 بت)
s_journal_dev	0xE4 (228)	__le32	4		رقم جهاز للملف قيْد الحوادث journal device في حالة تعيين علم ميزة قيد الحوادث الخارجي (قيمة 32 بت)
s_last_orphan	0xE8 (232)	__le32	4		بداية لائحة inodes اليتيمة من أجل الحذف
s_hash_seed[4]	0xEC (236)	__le32	16		البذرة أو القيمة الابتدائية للهاش باستخدام شجرة HTREE

s_def_hash_version	0xFC (252)	__u8	1	نسخة خوارزمية الهاش الاعتيادية المستخدمة في هاش الدليل (فهرسة الأدلة) وستكون إحدى دوال الهاش التشفيرية (قيمة 8 بت):			
				0x00	EXT2_HASH_LEGACY	Legacy	تراثي !
				0x01	EXT2_HASH_HALF_MD4	Half MD4	نصف دالة الهاش التشفيرية إم دي 4
				0x02	EXT2_HASH_TEA	Tea	خوارزمية التشفيرية الصغرى !
				0x03	EXT2_HASH_LEGACY_UNSIGNED	Legacy, unsigned	تراثي!، عدد صحيح لا يحمل إشارة
				0x04	EXT2_HASH_HALF_MD4_UNSIGNED	Half MD4, unsigned	نصف دالة الهاش التشفيرية إم دي 4، لا يحمل إشارة
				0x05	EXT2_HASH_TEA_UNSIGNED	Tea, unsigned	خوارزمية التشفيرية الصغرى!، عدد صحيح لا يحمل إشارة

s_jnl_backup_type	0xFD (253)	__u8	1	نوع النسخة الاحتياطية من قيد الحوادث (الاعتيادية) journal backup
s_desc_size	0xFE (254)	__le16	2	حجم توصيف مجموعات الكتل group descriptors، بعدد بايتات، في حالة تعيين علم ميزة INCOMPAT_64BIT

s_default_mount_opts	0x100 (256)	__le32	4	خيارات وصل نظام الملفات الاعتيادية (قيمة 32 بت) :	Default mount options :	
				0x0001	EXT4_DEFM_DEBUG	طباعة معلومات التنقيح عند وصل أو إعادة وصل نظام الملفات
				0x0002	EXT4_DEFM_BSDGROUPS	الملفات الجديدة تأخذ معرف مجموعة دليل الاحتواء gid (بدلا من معرف العملية الحالية fsgid)
				0x0004	EXT4_DEFM_XATTR_USER	دعم خصائص ممتدة توفرها مساحة المستخدم
				0x0008	(EXT4_DEFM_ACL	دعم قوائم التحكم بالنفاذ ACLs، معيار يونيكس (تصاريح نظام الملفات)
				0x0010	EXT4_DEFM_UID16	لا يدعم UIDs قيم 32-بت
				0x0020	EXT4_DEFM_JMODE_DATA	تنفيذ جميع البيانات والبيانات الوصفية إلى قيد الحوادث All data and metadata are committed to the journal
				0x0040	EXT4_DEFM_JMODE_ORDERED	تخليص جميع البيانات (من الصوان) إلى القرص قبل تنفيذ البيانات الوصفية إلى قيد الحوادث.
				0x0060	EXT4_DEFM_JMODE_WBACK	ترتيب البيانات غير محفوظ(غير محمي)؛ يمكن كتابة البيانات بعد كتابة البيانات الوصفية.
				0x0100	EXT4_DEFM_NOBARRIER	تعطيل كتابات الصوان إلى القرص write flushes (راجع آلية حواجز الكتابة BARRIER في EXT4)
				0x0200	EXT4_DEFM_BLOCK_VALIDITY	تعقب كتل البيانات الوصفية في نظام الملفات كي لا تستخدم ككتل بيانات. (هذا الخيار سيكون في حالة تمكين في 3.18)
				0x0400	EXT4_DEFM_DISCARD	تمكين دعم DISCARD، أين يتم إخبار جهاز التخزين عن الكتل التي أصبحت غير مستخدمة
				0x0800	EXT4_DEFM_NODELALLOC	تعطيل التخصيص المتأخر للكتل(راجع delayed allocation)

s_first_meta_bg	0x104 (260)	__le32	4	هوية أول مجموعة كتل وصفية في حالة تمكين ميزة meta_bg (قيمة 32 بت)
s_mkfs_time	0x108 (264)	__le32	4	زمن إنشاء نظام الملفات، بالثواني (توقيت يونكس)
s_jnl_blocks[17]	0x10C (268)	__le32	68	نسخة احتياطية من inode للقيد الحوادث (راجع حقل نوع النسخة الاحتياطية من قيد الحوادث)

دعم 64بت سيكون صالح في حالة تمكين ميزة EXT4_FEATURE_COMPAT_64BIT

s_blocks_count_hi	0x150 (336)	__le32	4	عدد الكتل الإجمالي (32 بت العليا)
s_r_blocks_count_hi	0x154 (340)	__le32	4	عدد الكتل المحجوزة (32 بت العليا)
s_free_blocks_count_hi	0x158 (344)	__le32	4	عدد الكتل الحرة (32 بت العليا)
s_min_extra_isize	0x15C (348)	__le16	2	جميع inodes يجب أن تملك # بايت على الأقل
s_want_extra_isize	0x15E (350)	__le16	2	inodes الجديدة يجب أن تحجز # بايت

s_flags	0x160 (352)	__le32	4	أعلام نظام ملفات متنوعة	Miscellaneous flags :		
				0x0001	EXT2_FLAGS_SIGNED_HASH	Signed dirhash in use	قيمة هاش دليل تحمل إشارة في الاستخدام
				0x0002	EXT2_FLAGS_UNSIGNED_HASH	Unsigned dirhash in use	قيمة هاش دليل لا تحمل إشارة في الاستخدام
				0x0004	EXT2_FLAGS_TEST_FILESYS	OK for use on development code	من أجل اختبار شفرة التطوير

				0x0010	EXT2_FLAGS_IS_SNAPSHOT	This is a snapshot image	هذه صورة زمنية للنظام snapshot
				0x0020	EXT2_FLAGS_FIX_SNAPSHOT	Snapshot inodes corrupted	inodes صور Snapshot فاسدة
				0x0040	EXT2_FLAGS_FIX_EXCLUDE	Exclude bitmaps corrupted	مصفوفات ثنائية فاسدة للصور Exclude bitmaps
s_raid_stride	0x164 (356)	__le16	2	وحدة شريطية في مصفوفة رديد RAID stride			
s_mmp_interval	0x166 (358)	__le16	2	منع الوصل المتعدد للنظام الملفات			# عدد ثواني انتظار فحص MMP.
s_mmp_block	0x168 (360)	__le64	8				# رقم كتلة بيانات حماية نظام الملفات من الوصل المتعدد MMP
s_raid_stripe_width	0x170 (368)	__le32	4	blocks on all data disks (N*stride)			حجم الشريط في مصفوفة رديد RAID stripe width
s_log_groups_per_flex	0x174 (372)	__u8	1	2 ^ s_log_groups_per_flex			حجم مجموعة الكتل المرنة (عدد مجموعات الكتل التي تشكل مجموعة flex_bg) وسيكون:
s_checksum_type	0x175 (373)	__u8	1	EXT2_CRC32C_CHKSUM			نوع خوارزمية تدقيق مجموع البيانات الوصفية. القيمة الوحيدة الصالحة هي 1 (crc32c)
s_reserved_pad	0x176 (374)	__le16	2	حشو/ محاذاة			
s_kbytes_written	0x178 (376)	__le64	8	عدد كيلوبايتات المكتوبة إلى نظام الملفات في فترة حياته (هذا مفيد في حالة تقدير كمية اهتراء الكتل على أقراص SSD (نتيجة دورات المسح المحدودة P/E cycles)			
s_snapshot_inum	0x180 (384)	__le32	4	صور زمنية للنظام snapshot		رقم inode الصورة النشطة (غير مستخدم في e2fsprogs / لينكس) Inode number of active snapshot	
s_snapshot_id	0x184 (388)	__le32	4			هوية تتابعية للصورة النشطة (غير مستخدم في e2fsprogs / لينكس) sequential ID of active snapshot	
s_snapshot_r_blocks_count	0x188 (392)	__le64	8			عدد الكتل المحجوزة للصورة النشطة للاستعمال مستقبلا (غير مستخدم في e2fsprogs / لينكس)	
s_snapshot_list	0x190 (400)	__le32	4			رقم inode بداية لائحة صور snapshot على القرص. (غير مستخدم في e2fsprogs / لينكس)	
s_error_count	0x194 (404)	__le32	4	الأخطاء	number of fs errors seen	عدد الأخطاء المنظورة	
s_first_error_time	0x198 (408)	__le32	4		first time an error happened	زمن وقوع أول خطأ، بعدد الثواني (توقيت يونكس)	
s_first_error_ino	0x19C (412)	__le32	4		inode involved in first error	inode المرتبط بأول خطأ	
s_first_error_block	0x1A0 (416)	__le64	8		block involved of first error	رقم الكتلة المرتبطة بأول خطأ	
s_first_error_func[32]	0x1A8 (424)	__u8	32		function where the error happened	اسم الوظيفة أين وقع الخطأ	
s_first_error_line	0x1C8 (456)	__le32	4		line number where error happened	رقم السطر أين وقع الخطأ	
s_last_error_time	0x1CC (460)	__le32	4		most recent time of an error	زمن أحدث خطأ، بعدد الثواني (توقيت يونكس)	
s_last_error_ino	0x1D0 (464)	__le32	4		inode involved in last error	inode المرتبط بأحدث خطأ	
s_last_error_line	0x1D4 (468)	__le32	4		line number where error happened	رقم السطر أين وقع أحدث خطأ	
s_last_error_block	0x1D8 (472)	__le64	8		block involved of last error	رقم الكتلة المرتبطة بأحدث خطأ	
s_last_error_func[32]	0x1E0 (480)	__u8	32		function where the error happened	اسم الوظيفة أين وقع أحدث خطأ	
s_mount_opts[64]	0x200 (512)	__u8	64	سلسلة ASCIIZ (ترميز أحرف) من أجل خيارات وصل نظام الملفات			
s_usr_quota_inum	0x240 (576)	__le32	4	حصص القرص	inode number of user quota file	رقم inode ملف حصة المستخدم	
s_grp_quota_inum	0x244 (580)	__le32	4		inode number of group quota file	رقم inode ملف حصة المجموعة	
s_overhead_blocks	0x248 (584)	__le32	4	العناقيدأو الكتل الفوقانية في نظام الملفات fs overhead blocks/clusters in (هذا الحقل دائما صفر، ويعني أن النواة تقوم بحسابه ديناميكياً)			
s_backup_bgs[2]	0x24C (588)	__le32	8	مجموعات الكتل التي تتضمن نسخ من superblock (في حالة تمكين ميزة sparse_super2)			
s_encrypt_algos[4]	0x254 (596)	__u8	4	خوارزمية التشفير المستخدمة. (راجع هذا) التي قد يصل عددها عند الاستخدام إلى 4 خوارزميات في أي وقت؛ شفرات الخوارزميات الصالحة مع أنماطها ستكون كالتالي:			
				0	EXT4_ENCRYPTION_MODE_INVALID		خوارزمية غير صالحة
				1	EXT4_ENCRYPTION_MODE_AES_256_XTS		AES إيه إي إس 256-بت في نمط XTS
				2	EXT4_ENCRYPTION_MODE_AES_256_GCM		AES إيه إي إس 256-بت في نمط GCM
				3	EXT4_ENCRYPTION_MODE_AES_256_CBC		AES إيه إي إس 256-بت في نمط CBC
				4	EXT4_ENCRYPTION_MODE_AES_256_CTS		AES إيه إي إس 256-بت في نمط CTS
s_encrypt_pw_salt[16]	0x258 (600)	__u8	16	سولت من أجل خوارزمية string2key (للتشفير) Salt used for string2key algorithm			
s_lpf_ino	0x268 (616)	__le32	4	موقع / رقم inode دليل lost+found			
s_prj_quota_inum	0x26C (620)	__le32	4	inode الذي يتعقب حصص القرص نوع project quotas			
s_checksum_seed	0x270 (624)	__le32	4	crc32c(~0, $orig_fs_uuid)			بذرة تدقيق المجموع المستخدمة في حسابات csum_seed.هذه القيمة في حالة تعيين metadata_csum
s_reserved[98]	0x274 (628)	__le32	392	حشو إلى نهاية الكتلة			
s_checksum	0x3FC (1020)	__le32	4	crc32c(superblock)			تدقيق مجموع Super block (حساب تدقيق مجموع بنية الكتلة العليا superblock يشمل أيضا FS UUID)

حجم الكتلة الإجمالي 1024 بايت (المصدر: ext4.wiki.kernel.org)

راجع بقية شرح تخطيط ext4 في كتيب EXT4

جدول توزيع الملفات المحسن / الممتد Extended File Allocation Table

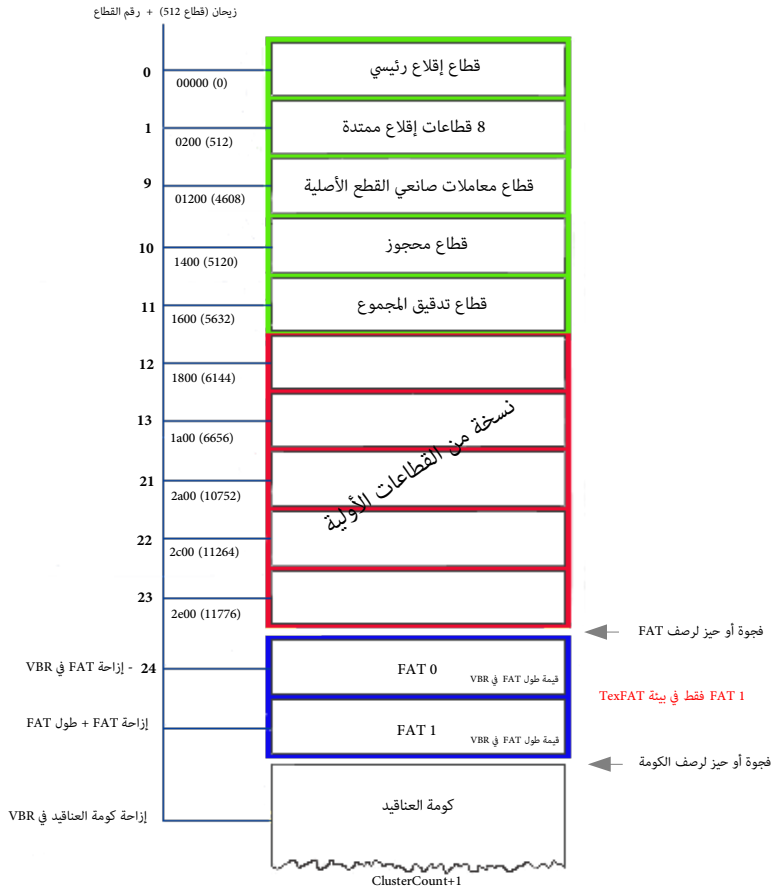
هذا نظام ملفات احتكاري من مايكروسوفت، (صدر عام 2006) يستخدم في الذواكر الوميضة الحديثة، مثل أقراص USB المحمولة، وبطاقة الذاكرة الرقمية المؤمنة. مثل ذواكر التخزين بتقنية SDXC خصوصا الأكبر من 32 جيجابايت. exFAT أيضا متوافق مع أقراص HDD و SSD المدمجة وغالبا ما يستخدم كالبديل للنظام ملفات NTFS (لأنه أقل فوقانية)، مع تجاوزه مشكلة حد حجم الملف 4 جيجابايت الموجود في FAT32..[42]

بنية وحدة تخزين exFAT

سجل إقلاع وحدة تخزين exFAT يتكون من ثلاث مناطق رئيسية هي منطقة الإقلاع، منطقة جداول FAT، ومنطقة البيانات(كومة العناقيد).

منطقة الإقلاع تتكون من 12 قطاع أولية، وأخرى احتياطية تأتي مباشرة بعد الأولية، وتتضمن:

- قطاع إقلاع رئيسي MBS مع معاملات BPB
- 8 قطاعات إقلاع ممتدة إضافية MEBS (من أجل برامج الإقلاع الكبيرة)
- قطاع محجوز (بدون استخدام حاليا)
- قطاع معاملات OEM (خاص بالعتاد)
- قطاع تدقيق المجموع (للتكامل بيانات القطاعات السابقة). [39]



• مصفوفة توزيع العناقيد (العنقود 2)	• Allocation Bitmap
• جدول المحارف الكبيرة (العنقود 3)	• Up-case Table
• الدليل الجذر (العنقود 4)	• Root Directory
• ملفات و أدلة	• Files & Directories

أول قطاع على وحدة تخزين ExFAT - (exfat super block)

```
dd if=/dev/sda3 of=exfat_sec_24 bs=512 count=24 | hexdump -C -n 512

          00 01 02 03 04 05 06 07  08 09 0A 0B 0C 0D 0E 0F  0123456789ABCDEF
0000  eb 76 90 45 58 46 41 54  20 20 20 00 00 00 00 00  |.v.EXFAT   .....|
0010  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
[Removed]
0040  3f 00 00 00 00 00 00 00  00 f8 9e 24 00 00 00 00  |?..........$....|
0050  00 08 00 00 00 4a 00 00  00 00 58 00 00 a0 9e 24  |.....J...X....$.|
0060  06 00 00 00 3e 54 95 60  00 01 00 00 09 01 80     |....>T.`........|
0070  00 00 00 00 00 00 00 00  33 c9 8e d1 bc f0 7b 8e  |........3.....{.|
0080  d9 a0 fb 7d b4 7d 8b f0  ac 98 40 74 0c 48 74 0e  |...}.}....@t.Ht.|
0090  b4 0e bb 07 00 cd 10 eb  ef a0 fd 7d eb e6 cd 16  |...........}....|
00a0  cd 19 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
00b0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
[Removed]
0100  0d 0a 52 65 74 69 72 65  7a 20 6c 65 20 64 69 73  |..Retirez le dis|
0110  71 75 65 ff 0d 0a 45 72  72 2e 20 64 69 73 71 75  |que...Err. disqu|
0120  65 ff 0d 0a 50 72 65 73  73 65 7a 20 75 6e 65 20  |e...Pressez une |
0130  74 6f 75 63 68 65 20 70  6f 75 72 20 72 65 64 82  |touche pour red.|
0140  6d 61 72 72 65 72 0d 0a  00 00 00 00 00 00 00 00  |marrer..........|
0150  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
[Removed]
01b0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 ff ff  |................|
01c0  ff ff ff ff ff ff ff ff  ff ff ff ff ff ff ff ff  |................|
[Removed]
01f0  ff ff ff ff ff ff ff ff  ff ff 00 14 22 55 aa     |............."U.|
0200
```

بايت	إزاحة	رمز تذكري	
3	0x00 (0)	JumpBoot	تعليمة القفزة JMP 0xEB7690 مطلوبة في أنظمة ملفات مايكروسوفت حتى وإن كان الجهاز لا يقبل الاقلاع

قيمة	وظيفة		بايت
EB	تعليمة القفزة	قفزة قصيرة SHORT JUMP	1
76	إزاحة نسبية		1
90	تعليمة لا عملية NOP		1

بايت	إزاحة	رمز تذكري	
8	0x03 (3)	FikeSystemName	هوية صانعي القطع الأصلية OEM ID. اسم أو لصيقة تشير إلى استعمال exFAT BPB الواقعة بين الحيد 40h و 77h

ست عشري	شفرة أسكي
45 58 46 41 54 20 20 20	"EXFAT "

بايت	إزاحة	رمز تذكري	
25	0x0B (11)	MustBeZero	كتلة معاملات DOS 3.31 BPB (في FAT32)
28	0x24 (36)		المعاملات الممتدة DOS 7.1 EBPB (في FAT32)

ماسك مكان
(53 بايت لا تستخدم، دائما 0)

exFAT BPB من الحيد 40h إلى 77h، تغطي المدخلات المعيارية FAT32 EBPB. وتعرف بالتوقيع " EXFAT" عند الحيد 003h، في هذه الحالة بايتات من الحيد 0x00B إلى 0x03F ستحمل قيم 0x00

بايت	إزاحة	رمز تذكري	
8	0x40 (64)	PartitionOffset	أول قطاع في وحدة التخزين (الإزاحة من MBR) أو تكون القيمة الاعتيادية.0 في الوسيط القابل للإزالة، وإن كانت 0 يجب تجاهلها. • مثال: القطاع الفيزيائي 63 = 00 00 00 00 00 00 00 0x3f
8	0x48 (72)	VolumeLength	حجم وحدة تخزين exFAT (عدد القطاعات) • مثال:614,266,880 قطاع = 00 00 00 00 24 9e f8 0x00
4	0x50 (80)	FatOffset	عنوان أول جدول FAT (القائمة الموصولة) (عدد القطاعات) (الإزاحة من VBR) • مثال: القطاع الفيزيائي 2048 = 00 00 08 00 0x00
4	0x54 (84)	FatLength	حجم FAT (القائمة الموصولة) (عدد القطاعات) • مثال:18,944 قطاع = 00 00 4a 00 x00
4	0x58 (88)	ClusterHeapOffset	عنوان بداية منطقة البيانات (بداية كومة العناقيد Custer heap) (عدد القطاعات) (الإزاحة من VBR) [40] [41] • مثال: القطاع الفيزيائي **22,528** = 00 00 58 00 0x00

مضمون بداية كومة العناقيد في exFAT.

• مصفوفة توزيع العناقيد (العنقود 2)	• Allocation Bitmap
• جدول المحارف الكبيرة (العنقود 3)	• Up-case Table
• الدليل الجذر (العنقود 4)	• Root Directory
• ملفات و أدلة	• Files & Directories

جدول Bitmap Allocation في كومة العناقيد.هو من سيتعقب حالة توزيع العناقيد

تحديد موقع العنقود المنطقي الأول:

المصفوفة الثانية Bitmap تبدأ مع العنقود 2. أي لإيجاد العنقود 0، نطرح عنقودين من قطاع بداية Bitmap.

مثال: 8192 = 00 00 20 00 0x00 (موقع قطاع العنقود 2) -2 = قطاع 8190

4		ClusterCount	عدد العناقيد [38] • مثال: 2,399,392 عنقود = 00 24 9e a0 0xa0
4	0x60 (96)	RootDirectoryCluster	أول عنقود في الدليل الجذر (الإزاحة من VBR) [36] [40] • مثال: 06 = 00 00 00 06 0x06 العنقود 6 (المحدد أعلاه) = 1536 (256 قطاع * 6 عنقود) + قطاع 22528 (بداية كومة العناقيد)— 2 عنقود (العنونة تبدأ عند العنقود 2) = القطاع 24062 إذن، عنقود بداية الدليل الجذر عند العنقود 6 وموقعه القطاع 24062
4	0x64 (100)	VolumeSerialNumber	الرقم التسلسلي للوحدة التخزين (الذي يولد باستخدام الوقت والتاريخ الجاري عند تهيئة وحدة التخزين)

			• مثال: الرقم المتسلسل: 543E-6095 = 0x3e 54 95 60
FileSystemRevision	0x68 (104)	2	رقم إصدارة نظام ملفات exFAT (بايت أعلى = رقم مراجعة رئيسية/هامة. و بايت أدنى = مراجعة ثانوية) • مثال:رقم هذه الإصدارة exFAT 1.00 = 0x00 01
VolumeFlags	0x6A (106)	2	أعلام (حالة) وحدة التخزين • مثال: 0x00 00 (16 بت) من أجل تعقب القائمة الموصولة والمصفوفة الثنائية النشيطتان، إلى جانب حالة وحدة التخزين. هذه القيم مخزنة في شكل بتات منفردة:

	معنى	علم	حجم	إزاحة	رمز تذكري
النسخة النشيطة FAT	أول FAT وأول Bitmap	بت 0	1 بت	0	ActiveFat [37]
	ثاني FAT وثاني Bitmap (فقط في TexFAT)	بت 1			
أخطاء وحدة تخزين	وحدة تخزين بدون أخطاء clean	بت 0	1 بت	1	VolumeDirty
	وحدة تخزين كثيرة الأخطاء dirty	بت 1			
فشل الوسيط	لم يعلن عن أي فشل أو تم وسمها سابقا كعناقيد تالفة	بت 0	1 بت	2	MediaFailure
	فشل بعض عمليات القراء/الكتابة	بت 1			
تصفير	بلا معنى	---	1 بت	3	ClearToZero
محجوزة	بلا معنى	---	12 بت	4	Reserved

BytesPerSectorShift	0x6C (108)	1	عدد بايتات لكل قطاع (قوة العدد اثنين أي 1، 2، 4، 8، الخ).القيمة الأدنى 2^9 (512 بايت لكل قطاع) والأقصى 2^{12} (4096 بايت لكل قطاع) • مثال: حجم القطاع 2^9 (512 بايت) = 0x09
SectorsPerClusterShift	0x6D (109)	1	عدد القطاعات لكل عنقود (قوة العدد اثنين) القيمة الأدنى 2^0 (1 قطاع لكل عنقود) والأقصى 2^{25}، لذا أقصى حجم للعنقود هو 32 ميغابايت. • مثال: 2^8 = 256 قطاع لكل عنقود = 0x08
NumberOfFats	0x6E (110)	1	عدد نسخ FAT على وحدة التخزين، دائما 1 (أو تكون 2 في بيئة TexFAT) • مثال: 0x01 = 1
DriveSelect	0x6F (111)	1	رقم القرص (يستخدم من قبل نداء المقاطعة Extended INT 13h)، سيكون دائما 0x80
PercentInUse	0x70 (112)	1	نسبة استعمال منطقة البيانات، (0xff = غير متوفر) النسبة المئوية للعناقيد المخصصة مع تدوير إلى عدد صحيح، بدون كسور. مثال: 0x01 = 1%
Reserved	0x71 (113)	7	محجوزة، دائما 0 مثل: 0x00 00 00 00 00 00 00
BootCode	0x78 (120)	390	شفرة إقلاع تنفيذية (مثال الطرح أعلاه— مع النسخة الفرنسية)
BootSignature	0x1FE (510)	2	توقيع (0xAA55 = 55AA)
ExcessSpace	0x200 (512)	2^BytesPerSectorShift - 512	لا تستخدم (إذا كان حجم القطاع أكبر من 512 بايت، سيكون هذا حشو إضافي بعد توقيع الإقلاع)

بنية قطاع الإقلاع الممتد— (8 قطاعات إقلاع ممتدة إضافية (MEBS)

```
0200 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
[Removed]
03f0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 55 aa |..............U.|
```

ExtendedBootCode	0x00 (0)	2^BytesPerSectorShift - 4	شفرة إقلاع ممتدة (شفرة تتبع النظام)
ExtendedBootSignature	2^BytesPerSectorShift - 4	4	توقيع قطاع الاقلاع الممتد 0xAA550000

في القطاعات الممتدة، شفرة الإقلاع تشغل كامل القطاع باستثناء توقيع 4 بايت في كل قطاع. لكن إذا لم يستخدم قطاع الإقلاع الممتد، سيكون معبأ بقيم 0x00 مع وجود توقيع 000055AA (أنظر للطرح)

قطاع معاملات صانعي القطع الأصلية OEM

```
1200 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
[Removed]
13f0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
```

إزاحة	بايت	
0x00 (0)	48	معاملات [0] حاليا هذه المعاملات العميمة

رمز تذكري	إزاحة	حجم	بنية سجل معاملات OEM
ParameterType	0x00 (0)	16	نوع المعامل، --- صانعي القطع الأصلية هو من يحدد رقم معرف GUID
ParameterValue	0x10 (16)	32	قيمة المعامل --- مخصصة من قبل صانعي القطع الأصلية OEM

```
#define OEM_FLASH_PARAMETER_GUID 0A0C7E46-3399-4021-90C8-FA6D389C4BA2
struct
{
 GUID OemParameterType; //Value is OEM_FLASH_PARAMETER_GUID
 UINT32 EraseBlockSize; //Erase block size in bytes
 UINT32 PageSize;
 UINT32 NumberOfSpareBlocks;
 UINT32 tRandomAccess;  //Random Access Time in nanoseconds
 UINT32 tProgram;   //Program time in nanoseconds
 UINT32 tReadCycle; //Serial read cycle time in nanoseconds
 UINT32 tWriteCycle;    //Write Cycle time in nanoseconds
 UCHAR Reserved[4];
}
FlashParameters;
```

قيم الصفر = غير محددة

- - -	- - -		- - -
معاملات [9]		48	0x1B0 (432)
محجوزة		2^BytesPerSectorShift - 480	0x01E0 (480)

معاملات OEM (خصائص الأجهزة مثل ذاكرة فلاش flash devices)، يتجاهلها نظام ويندوز، لكن يمكن أن تستخدمها تطبيقات OEM. صانعي القطع الأصلية هم من يحدد معاملاتهم مع معرفاتها GUIDs. حقول المعاملات التي لا تستخدم يجب وصفها كذلك بالدالة GUID_NULL في حقل ParameterType. أي OemParameterType وهذه البنية إن كانت موجودة، يجب المحافظة عليها أثناء تهيئة exFAT، باستثناء حالة المسح الأمن للوسيط

بنية معاملات OEM

```
struct
{
   struct
      {
         //If the GUID is all zeroes, it indicates the parameter is undefined.
         GUID OemParameterType;
         //OEM-specific parameter structure
         UCHAR OemParameters[32];
      }
   OemParameterArea[10];
   UCHAR Reserved[32];
}
OEM_PARAMETERS;
```

القطاع المحجوز

القطاع لا يستخدم / محجوز، ومصفر. إزاحة هذا القطاع ستكون من 1400 إلى 15ff. (في حلة قطاع 512 بايت)

قطاع تدقيق مجموع (32-بت) VBR Checksum

القطاع 12 يتضمن 4-بايت مكررة. تشكل تدقيق مجموع 11 قطاع سابقة. تدقيق المجموع يستثنى حقول VolumeFlags و PercentInUse (متطايرة)، تدقيق المجموع مكرر حتى نهاية القطاع. مرات التكرار يعتمد على حجم القطاع.

```
1600 C9 D0 18 8B C9 D0 18 8B C9 D0 18 8B C9 D0 18 8B | ÉÐ.‹ÉÐ.‹ÉÐ.‹ÉÐ.‹|
[Removed]
17f0 C9 D0 18 8B C9 D0 18 8B C9 D0 18 8B C9 D0 18 8B | ÉÐ.‹ÉÐ.‹ÉÐ.‹ÉÐ.‹|
```

```
UINT32 BootChecksum (const unsigned char data[], int bytes)
{
 UINT32 checksum = 0;
}
 for (int i = 0; i < bytes; i++)
 {
  if (i == 106 || i == 107 || i == 112)
   continue;
  checksum = (checksum << 31) | (checksum >> 1) + data[i];
 }
 return checksum;
}
```

بقية القطاعات من 12 إلى 23 تتضمن نسخة احتياطية كاملة من 12 قطاع الأولى على وحدة التحزين (القسم)

من ميزّات exFAT (مواصفة 1.00) :

- حجم القطاع من 512 إلى 4096 بايت
- حجم العنقود يمكن أن يصل إلى 32 ميبايت
- FAT32 يملك من 32-بت، يستخدم منها 28 بت (4-بت العليا من عنقود 32 بت محجوزة) و exFAT يملك من 32-بت، يستخدم منها 32 بت (تنبيه: تسمية FAT64 غير صحيحة /غير رسمية).
- الأدلة الفرعية يمكن أن تصل إلى حجم 256 ميبيبايت.
- يملك بعض ميزّات NTFS، وأسرع، وأقل فوقانية overhead.
- يملك قطاع من أجل معاملات صانعي القطع الأصلية OEM (يمكن تخصيص نظام الملفات وفقا للخصائص الجهاز)
- حجم وحدة التحزين الأقصى يصل حاليا ≈ 128 يتابايت (حد نظري) ويوصى بالحجم 512 تيرابايت (حد فعلي) (تنبيه: قسم exFAT في MBR مقيد بـ 4 بات في عدد القطاعات)
- حد حجم الملف 1 - 16 إكسابايت (حد نظري) (أي أكبر من حجم وحدة التخزين!) أو 512 تيرابايت (حد فعلي).
- عدد الملفات في كل دليل (فرعي) يصل إلى 2,796,202 ملف،
- إزالة مدخلات الدليل الفيزيائية "." و ".." التي تظهر في الأدلة الفرعية
- العدد الأقصى للملفات على وحدة التخزين، حوالي 2^{32}-11
- عدد المحارف الأقصى في أسماء الملفات 255 محرف UTF-16،
- أسماء ملفات يونيكود و أسماء وحدات التخزين،
- exFAT لا يدعم أسماء الملفات القصيرة SFN (أي صيغة 8.3)
- دعم معيار نظم ملفات العمليات،TexFAT (اختياري في ويندوز سي إي WinCE)
- زيحان من أجل رصف حدود جدول FAT وزيحان أخر لرصف حدود منطقة البيانات (أنظر للخطاطة)
- (اختياري) دعم قائمة التحكم بالنفاذ ACL (أذون نظام الملفات) (فقط في ويندوز سي إي 6 Windows CE)
- ثلاثة مجاميع للتدقيق المجموع لضمان سلامة أو تكامل البيانات الوصفية
- exFAT ليس من نوع نظام الملفات المزود بقيد حوادث، لكنه يستخدم جدول توزيع ملفات واحد مع مصفوفة للمساحة الفارغة/الحرة. يسمح باستعادة نظام الملفات في حالة سحب ejected الوسيط أثناء الكتابة (خصوصا مع الأقراص القابلية للإزالة)، أيضا خيار TexFAT أضاف دعم للمصفوفات والجداول الاحتياطية، لكنها غالبا ليست مدعومة في الأنظمة.

ملاحظات

1. ^ في قطاعات الإقلاع، التوقيع الذي عند الحيد 1FEh+ سيكون AAh 55h حيث 55h عند 1FEh+ و AAh عند 1FFh+.وبما أن نهوي صغير little-endian هي الطريقة المستخدمة في تمثيل ترتيب البيانات في الأجهزة المتوافقة مع أنظمة أي بي أم، IBM PC ، هذه يمكن كتابتها بكلمة 16-بت AA55h في برامج معالج أنظمة x86 (لحظ ترتيبها المعكوس)، بينما تكتب بشكل 55AAh في برامج المعالجات الأخرى بطريقة نهوي كبير big-endian. لأن هذه الطرق في التمثيل قد تم الخلط بينها في الكتب وفي وثائق المرجع الأصلي والرسمي من مايكروسوفت، النص المترجم من الموسوعة يستخدم طريقة البايت المرتكز على الإزاحة في تمثيل البيانات على القرص.

أمثلة أخرى : هذان الملفين من أداة سطر الأوامر لمراقب الشبكة Tcpdump في إنتيل و سن ميكرو سيستم تعرض الاختلاف بين نهوي-كبير ونهوي-صغير مع نفس البيانات في يونكس.

سولاريس على جهاز سن ميكرو سيستم (نهوي-كبير)		لينكس على جهاز إنتيل (نهوي-صغير)	
00020004	A1B2C3D4	02000400	D4C3B2A1
00000001	00000044	01000000	60000000
0004BFF0	3EBCBA2D	46C30500	2DBABC3E

2. ^ في وحدة تخزين FAT32، الدليل الجذر جزء من منطقة البيانات وليس منطقة ثابتة منفصلة على القرص كما هو الحال في وحدة تخزين FAT16 في ويندوز 95 . أيضا وحدة تخزين FAT32 تملك حقل جديد يدعى رقم عنقود بداية الدليل الجذر في كتلة FAT32 BPB.
3. ^ على عكس سجلات الاقلاع السابقة، الشفرة في هذا السجل تستخدم في الذاكرة بايت (تعليمة لا عملية 90h) عند عنوان 0000:7C02h كعلم يدل على استخدام أو عدم استخدام وظيفة قراءة قطاعات القرص (أي الوظيفة 42h التي هي امتداد في نداء المقاطعة INT 13).
4. ^ tight loop أو busy-loop هي دورة حلقية أو متكررة حلقية تنفذ دون إصدار أية موارد إلى البرامج الأخرى أو نظام التشغيل. علما أن الحلقات اللا نهائية endless loop، infinite loop غالبا ما تكون tight، لكن tight loop نادرا ما تكون لا نهائية. أما JR فهو تسجل القفزة.
5. ^ 32768 في إصدارة TOS4.0 على جهاز فالكون Falcon (لكن رسميا يدعم فقط 16384).
6. ^ في (TOS < 1.04) حجم القسم الأقصى = 256 ميغابايت (2^{15} * 8192)، و في TOS 4.x حجم القسم الأقصى = 2 جيجابايت (2^{16} * 32768)
7. ^ في بيئة أتاري يستخدم غالبا مصطلح تمهيد/تهيئة "Initialization" بدلا من مصطلح تهيئة "Formatting" المستخدم في بيئة الحاسوب الشخصي.
8. ^ رغم إمكانية تبادل الملفات بين الجهازين MSX ↔ PC (المتوافقة مع الحاسوب الشخصي)، جهاز أو محاكي MSX سوف يعلق إذا حاولت إقلاع نظام MSX-DOS من قرص مرن مهيئ في نظام MS-DOS في الحاسوب الشخصي، لأن قطاع الاقلاع وشفرة الاقلاع مختلفة عن شفرة قطاع إقلاع القرص المهيئ والمستخدم في جهاز MSX. (لاختلاف بنية العتاد والمعالج). مع هذا لا توجد أية اختلافات أخرى، باستثناء أن MSX DOS 2 يستخدم منطقة نسخة احتياطي في FAT لاستعادة الملفات المحذوفة، عن طريق الأمر UNDEL.

الأقراص المرنة 360 كيلوبايت (ذات الوجه الواحد) يمكن أيضا قرأتها في أجهزة الحاسوب الشخصي. لكن نتيجة لأن واصف الوسيط للقرص 360 كيلوبايت في MSX مساوي للقرص الثابت في الحاسوب الشخصي، هذا الأخير لن يتعرف على القرص المرن MSX. لكن يمكن حل هذه المشكلة بتغيير شفرة واصف الوسيط لتتناسب مع الحاسوب الشخصي مع القرص المرن 360 كيلوبايت (راجع ذلك في حزمة CJS MSX2 emulator. وصفحة الأسئلة).

9. ^ بنية قطاع الاقلاع تغيرت قليلا في أقراص MSX-DOS2 مقارنة بظام MSX-DOS1. من أجل تمييز الأقراص تم إضافة هوية وحدة التخزين volume ID. نظام MSX-DOS2 إذا لم يجد هذه المدخلة لهوية وحدة التخزين على القرص، سوف يعرض رسالة الخطأ "Wrong version of MSX-DOS". النظام يفترض أن القرص ليس قرص DOS2 وهذا يعني أن وظائف (ميزات) مثل استعادة الملفات المحذوفة والذاكرة المؤقتة cache لن تعمل تماما. ولأن هذا يمكن أن يسبب مشاكل، MSX-DOS2 يتوقع من المستخدم معرفة إعدادات وضع expert. واستخدام الأمر : SET EXPERT = ON. لكن هذا الحل لا يمكن وظيفة استعادة الملفات المحذوفة، ولحل المشكلة استخدم طريقة FIXDISK أو FIXBOOT المتوفر في موقع BiFi، والمستخدمة في الأساس مع أقرص IDE.. يمكن ترقية القرص إلى بنية MSX-DOS2 باستخدام برنامج FIXDISK الموجود في MSX-DOS2، مثال :FIXDISK A.
10. ^ المعرف label يستخدم في تأشير موقع في برنامج ليكون وجهة العبارة/التعليمة JMP (أي القفز من مكان إلى آخر في البرمجيات).
11. ^ أصل التسمية BPB !

> عكس ما قد تعتقد، كلمة بيوس BIOS في جملة BIOS parameter block ليست إشارة إلى نظام الإدخال والإخراج الأساسي التقليدي المعروف اختصارا باسم BIOS (واجهة البرنامج الثابت). ببساطة هذا الأخير في الحاسوب الشخصي يجهل كليا حقيقة معاملات BPB.
>
> جذور هذه التسمية تعود إلى تصميم نظام تشغيل مايكروسوفت/بي سي دوس، الإصدار 2.0 (المستمد من تصميم CP/M) حين كان نظام التشغيل يقسم إلى نصف علوي، يتعامل مع نداءات النظام وتحريد الملفات والأدلة، ونصف آخر سفلي، يتعامل مع مشغلات العتاد وأساسيات الوصول الفيزيائي إلى أجهزة التخزين بالنفاذ المباشر DASD.
>
> النصف العلوي كان يدعى نظام تشغيل القرص الأساسي أو اختصارا بدوس BDOS، والنصف السفلي كان نظام الإدخال والإخراج الأساسي أو اختصارا بيوس BIOS. هذان الجزءان من نظام م.س-دوس/بي سي دوس كنا يحتفظ بهما في ملفين منفصلين (على شكل صور) على القرص، في كافة إصدارات م. س.دوس/بي سي دوس المختلفة حتى الإصدار 5. وفي بي سي دوس، ودي آر دوس، كانت تسمى على التوالي IBMDOS.COM (النواة) و BMBIO.COM (شفرة تهيئة ومشغلات)، أسماء الملفات هذه تعكس مباشرة أسماء مكونات نظام التشغيل. نستنتج من ذلك أن كلمة BIOS في BPB هي إشارة فقط إلى النصف السفلي في مايكروسوفت/بي سي/آر دي دوس.

12. ^ في المعايير الدولية (خرطوشه) القرص المرن، flexible disk cartridge أو FDC هو الاسم الرسمي للقرص المرن.
13. ^ متجه المقاطعة interrupt vector مكان في ذاكرة الحاسوب يخزن عناوين روتينات الثانوية التي تم تنشيطها عند إغلاق الحاسوب.
14. ^ محمل ابتدائي ؟ Initial Program Loader / IPL : في سجل الاقلاع الرئيسي، المساحة المخصصة لمحمل الاقلاع 446 بايت لا يمكن أن تتضمن كامل شفرة الاقلاع، لهذا تحتل هذه المنطقة شفرة ابتدائية تدعى اختصارا IPL (وتعني وظيفيا : محمل ابتدائي للشفرة). من أسماء هذه الشفرة في لينكس stage1 و boot.img. في أنظمة ويندوز هذه الشفرة تدل نظام BIOS إلى قطاع إقلاع ويندوز.

وفي لينكس تحمل شفرة core.img أو stage1.5 التي عادة تكون في المسار الأول من القرص مباشرة بعد سجل الاقلاع الرئيسي، أو تحمل مباشرة الشفرة الرئيسية core.img أو stage2 من نظام الملفات.

15. ^ المرآوية Mirror/Mirroring (فعل/اسم) تعي عمل نفس النسخة من البيانات، لكنها عادة تشير إلى :
    - مرآوية القرص أو RAID 1، التي تعني كتابة نفس البيانات على جميع الأقراص.
    - موقع مرآة، موقع أرشفي، موقع على الأنترنت، الذي يعيد نشر المعلومات حرفيا من موقع (منشأ) آخر.
        - مرآة تحميل؛ تشير إلى بروتوكول FTP. أو مرآة موقع يستخدم في توزيع الملفات الكبيرة مثل لينكس أو البرمجيات الحرة أخرى.

16. ^ نظام ملفات BigFAT / BIGDOS / FAT16B: الاسم التقني لنظام الملفات هذا هو FAT16B. (حرف FAT16B = BigFAT / Big = B) وهو نسخة أحدث من بنية نظام الملفات الأصلي FAT16). أداة مايكروسوفت DSKPROBE تشير إلى نوع القسم 0x06 باسم BigFAT، بينما بعض النسخ القديمة من FDISK تصف النوع باسم BIGDOS. نظام FAT16B يستخدم صيغة أحدث من كتلة معاملات BPB مع حجم مدخلة قطاع 32-بت.

17. ^ هذا القطاع تقنيا يعرف ببنية BIGFATBOOTFSINFO ؛ وتعني قطاع معلومات نظام الملفات في سجل إقلاع قسم BigFAT. بعض المصادر تذكر أن "BigFAT "هو الاسم الأصلي لنظام FAT32.

18. ^ التفرع Branch أو القفز jump ؟ : الاختلاف الرئيسي بين الاثنين سيكون في الذاكرة وفي زمن المعالجة. تعليمة BRA أسرع من تعليمة JMP، وأصغر حجم، هذا يعني الاقتصاد في الوقت والذاكرة. لكن تعليمة BRA.S أو حتى BRA.W لا يمكنها الوصول إلى أماكن معينة، عكس تعليمة JMP. محرف s في "bra.s" يعني قصيرة short. الحجم الأخر المتوفر هو "bra.w" حيث "w" تعني كلمة word (أي 2 بايت) (معالج موتورولا).

19. ^ الفرق بين Intra segment و Inter segment ؟ : قفزات القطعة الداخلية Intra segment jumps دائما تكون بين العبارات (أو الأوامر الأحادية) statements. (داخل قطعة شفرة واحدة). قفزات القطعة البينية Inter segment jumps تستطيع نقل التحكم إلى أمر أحادي statement داخل قطعة شفرة CS مختلفة (بين أكثر من قطعة واحدة). القفزات القصيرة والقريبة غالبا ما تدعى Intra segment jumps، بينما القفزات البعيدة تدعى Inter segment jumps. بالمناسبة، قطعة الشفرة CS تدعى أيضا قطعة البرنامج أو مقطع البرنامج (في ترجمات عربية أخرى)

20. ^ كتلة BPB تدعى أيضا باسم Big FAT BIOS Parameter Block ويرمز لها BF_BPB. علما أن Big FAT تشير إلى نوع القسم 06h. نظام ملفات FAT16B (بعض المصادر تذكر أن BigFAT هو الاسم الأصلي لنظام ملفات FAT32).

21. ^ سجل الاقلاع (قطاع الاقلاع) أحيانا يدعى سجل إقلاع ممتد Extended Boot Record، نظرا لأن سجل الاقلاع القديم في FAT16 كان يملك قطاع واحد فقط.

22. ^ hard wired, Hard coded : (مفردة تخصصية) هي قيمة بيانات أو إجراء تم كتابته مباشرة في برنامج، غالبا في عدة أمكان، بحيث لا يمكن تعديلها 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.

23. ^ مهيئ مضيف ، مكيف مضيف host adapter : جهاز يربط الوحدة الملحقة (عن طريق ممر SCSI) بالحاسوب الرئيسي، عادة، في شكل بطاقة توسعة. الجهاز يدعى أيضا متحكم controller، ومكيف ناقل مضيف host bus adapter. كلمة المضيف HOST تشير إلى الحاسوب الرئيسي.

24. ^ LSB/MSB : البت ذو القيمة الأدنى، الخانة الأقل أهمية (LSB) أو (least significant bit) : بت (على اليمين) ويسمى بت منخفض low bit (بت أدنى في البايت). بمعنى : صفر بت (لأن ترقيم البت يبدأ من 0 ثم يزداد بواحد في كل موضع بت لاحقا)، هو بت رقم ثنائي يعطي رقم أحاد ones ، وهو 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 least significant bit.

البت ذو القيمة الأعلى، الخانة الأكثر أهمية (MSB) : آخر بت (على اليسار) ويسمى high bit (بت أعلى في 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 Most significant bit.

25. ^ نداء النظام System call : ألية تستخدم من قبل البرنامج لطلب خدمة من نظام التشغيل. نداءات النظام غالبا ما تستخدم تعليمة لغة آلة خاصة تجعل المعالج يغير نمط التشغيل (مثل، نمط المشرف s upervisor mode، النمط المحمي protected mode). هذا يسمح لنظام التشغيل القيام بإجراءات محدودة مثل النفاذ إلى العتاد أو وحدة إدارة الذاكرة MMU. (راجع System call في دليل MSX2).

26. ^ لماذا نستخدم هنا تعبير قسم وليس وحدة تخزين ! : ويندوز XP/2000 يخزن نسخة من كل سجل إقلاع وحدة تخزين في القطاع الأخير في القسم !. في هذا الحالة سيكون حجم وحدة التخزين أصغر بقطاع واحد من القسم الذي يضمها ؛ "عدد القطاعات الإجمالي" في القسم NTFS في جدول أقسام MBR/EBR دائما أكبر بقطاع واحد من "عدد القطاعات الإجمالي" في وحدة التخزين الموجود في سجل إقلاعها VBR. رغم أن المصطلحان قسم (أولي) و وحدة تخزين غالبا ما تكون مترادفة، لكن في هذه الحالة لا يحملان تقريبا نفس المعنى.

27. ^ سلسلة محارف متبوعة ببايت صفر وحيد، تستخدم في عدة لغات برمجة. وتعرف بسلسلة منتهية بصفر Null-terminated string.

28. ^ رغم أن مايكروسوفت، دائما تستخدم في سجلات الاقلاع، تعليمة القفزة القصيرة 2-بايت، Short JMP التي تبدأ بالبايت EB. يمكن بسهولة أيضا استخدام تعليمة القفزة القريبة، المباشرة Direct JMP التي تبدأ بالبايت E9 وتحتاج 2 بايت إضافية للإزاحة النسبية. مثلا تعليمة، EB 58 90 أو EB 52 90 التي تظهر في المثال (2-بايت للقفزة القصيرة زائد 90h من أجل تعليمة لا عملية NOP) يمكن استبدالها في سجل الاقلاع بثلاثة بايت E9 57 00 أو E9 51 00 (والثلاثة بايتات ستكون جزء من التعليمة القريبة، عكس تعليمة القفزة القصيرة).

29. ^ في وحدات التخزين التي تم فيها حذف وإضافة عدة ملفات، خصوصا، وحدات التخزين التي على وشك أن تنفذ فيها مساحة التخزين، جدول الملف الرئيسي MFT يمكن أن يتحول من مكانه.

30. ^ قطاع الاقلاع NTFS يحمل 15 قطاع إضافية في الذاكرة تتضمن شفرة الاقلاع؛ رغم أن القطاعات الثمانية الأخيرة منها تتضمن فقط أصفار !

31. ^ قرص ميكرو فلوبي أو الدقيق microfloppy disk : هو قرص مرن قطره أقل من 5¼ بوصة، 13.3 سنتيمتر (عادة، 3½ بوصة، 8.9 سنتيمتر).

32. ^ منطقة القطاعات المخفية : هذه القيمة من المفترض أن تكون عدد القطاعات الفيزيائية على القرص التي تسبق القطاع الأول لوحدة التخزين؛ هذا يفسر لماذا مدخلة القسم الأول في جدول أقسام القرص تملك القيمة 63، هذه القيمة ستكون قي حالة القسم الأولي الأول في القرص الثابت (من 1-0-0 CHS إلى 63-0-0 CHS) أو القرص المنطقي الأول في القسم الممتد (لأن كل وحدة تخزين في القسم الممتد يسبقها سجل إقلاع ممتد خاص)، لكن في حالة 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 أو المحجوزة لأول قسم ارتفع إلى 2048 قطاع بدل 63. وكذلك الحال مع برامج تقسيم القرص في الأنظمة الأخرى.

33. ^ Jumps with Register Operands و Indirect Jumps using an index.

قفزة مع استخدام معاملات التسجيل :

- القفزة يمكنها استخدام أيضا التسجيل 16-بت أو 32-بت كمعامل.
  - وتنصب آليا كقفزة غير مباشرة Indirect Jump.
  - عنوان القفزة يكون في التسجيل تحدده تعليمة القفزة.
- على خلاف الإزاحة المصاحبة للقفزة القريبة Near Jump، محتويات التسجيل تنقل (تحمل) مباشرة إلى مؤشر التعليمة.
- القفزة الغير مباشرة Indirect Jump لا تضاف إلى مؤشر التعليمة.
- مثال : JMP AX، تنسخ محتويات التسجيل AX إلى IP.
  - هذه تسمح بقفزة إلى أي موقع ضمن قطعة الشفرة CS الحالية.

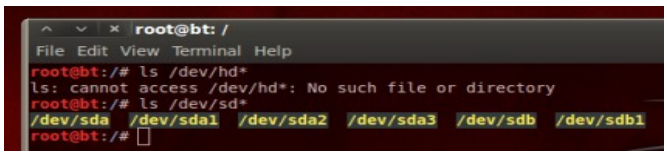
القفزة الغير مباشرة باستخدام الفهرسة :

- تعليمة القفزة تستخدم أيضا أقواس مربعة [ ] كشكل من العنونة للنفاذ مباشرة إلى جدول القفزة (يتضمن سلسلة من تعليمات القفزة/التفرع اللامشروطة).
- جدول القفزة يمكن أن يتضمن عناوين الإزاحة للقفزات القريبة الغير مباشرة، أوعناوين الإزاحة والقطعة للقفزات البعيدة الغير مباشرة.
  - هذه تعرف أيضا باسم القفزة الغير مباشرة المزدوجة Double-indirect Jump إذا قفزة التسجيل دعيت قفزة غير مباشرة Indirect Jump.
- المجمع يفترض أن القفزة قريبة Near Jump ما لم يشر الأمر التوجيهي FAR PTR إلى تعليمة قفزة بعيدة Far Jump.

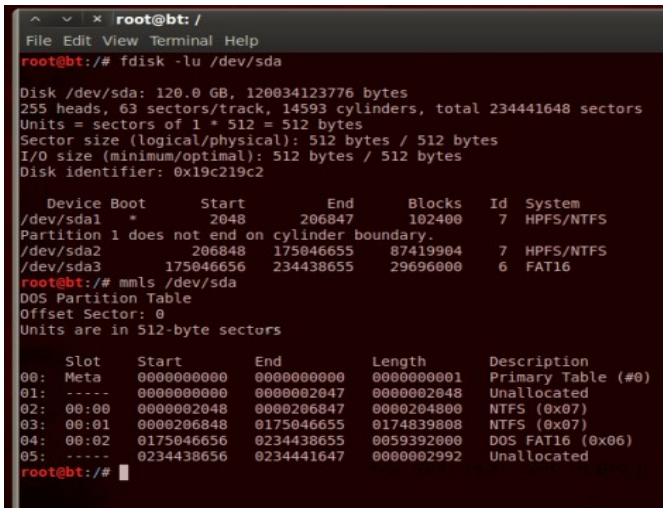
34. ^ علامة النجمة * في الطرح تدل على القيم المحذوفة (الأسطر المكررة) (في هذا المثال كانت القيم "أصفار").لعرض الشفرة في لينكس يمكنك استخدام هذه الأوامر.

طرح بدون القيم المكررة :	# hexdump -C file
طرح كامل :	# hexdump -Cv file
طرح القطاع الأول فقط (إن كان الملف كبير)	# hexdump -Cv -n 512 file

35. ^ أ ، ب استخلاص وتحليل معلومات نظام ملفات على الأقراص (الموصولة بجهاز الحاسوب) باستخدام أوامر طرفية لينكس [5]:

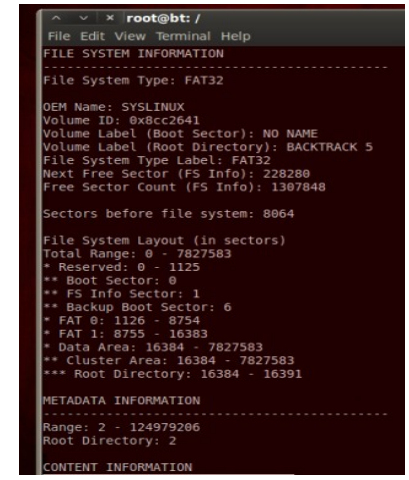


في هذا المثال 3 أقسام على القرص sda وواحد على القرص الثاني sdb (قرص ذاكرة فلاش) نوع الجهاز كان SATA أو SCSI



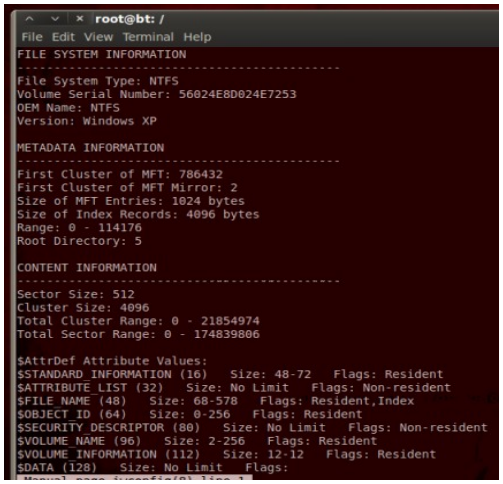
- معلومات عن القرص/الأقسام مثل قطاعات البداية/النهاية قيم CHS، عدد الكتل، حجم القطاع المستخدم، معرف القرص، العدد الإجمالي للقطاعات، ونوع نظام الملفات
  - الخيار -L لسرد جدول الأقسام المستخدم في الجهاز المحدد. والخيار u- لعرض الأحجام بالقطاعات بدلا من الأسطوانات
- أداة mmls تعرض تخطيط وحدات التحزين (بعدد القطاعات) بما في ذلك البيانات المخفية على القرص

أول سطرين 00 و 01 تشير إلى قطاع MBR و المساحة الغير مستعملة (غير مخصصة) بين قطاع MBR وأول قسم على القرص



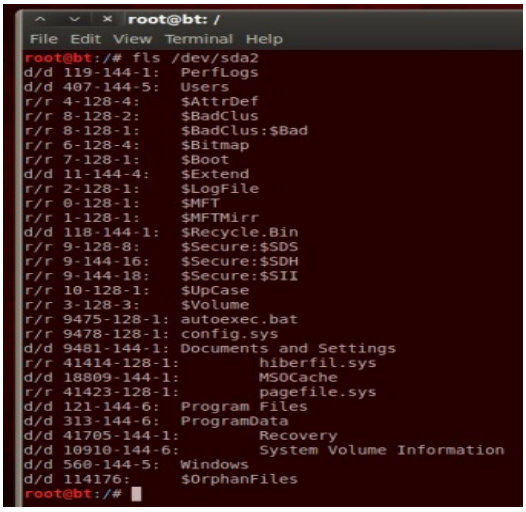
fsstat /dev/sdb1 | less

تخطيط وحدة التخزين FAT32 (على قرص ذاكرة فلاش - BackTrack/Kali)



fsstat /dev/sda2 | less

تخطيط وحدة التخزين NTFS (القسم الثاني على القرص الثابت)



عرض أسماء الملفات والأدلة في نظام الملفات، بما فيها أسماء الملفات المحذوفة مؤخرا

36. ^ لإيجاد موقع بداية القطاع الأول للدليل الجذر (العنقود الأول)، نبحث عن إزاحة العنقود النسبية إلى الموقع المحدد للعنقود 2.

- عنقود 261 = 0x05 01 00 00 (بداية الدليل الجذر)
- في هذا المثال: قطاع واحد = عنقود واحد (1 قطاع = 1 عنقود)
- العنقود 2 يبدأ في قطاع 8192. (0x20 00 00 00) (بداية Bitmap)
- العنقود 261 = 0x0105 (المحدد أعلاه) =261 قطاع = (1 قطاع × 261 عنقود) + قطاع 8192 (بداية لكومة العناقيد)— 2 عنقود (العنونة تبدأ عند العنقود 2) = القطاع 8451.
- إذن، عنقود بداية الدليل الجذر عند العنقود 261 وموقعه القطاع 8451

مثال ثاني:

- عنقود 70 = 0x46 00 00 00 (بداية الدليل الجذر)
- في هذا المثال: قطاعيين = عنقود واحد (2 قطاع = 1 عنقود)
- العنقود 2 يبدأ في القطاع 4224 (0x80 10 00 00) (بداية مصفوفة ثنائيات توزيع العناقيد Bitmap)
- العنقود 70 = 0x46 (المحدد أعلاه) = 140 قطاع (2 قطاع × 70 عنقود) + 4224 (موقع بداية منطقة البيانات)— 2 عنقود (4 قطاعات) = القطاع 4360
- إذن، عنقود بداية الدليل الجذر عند العنقود 70 وموقعه القطاع 4360

مثال ثالث :

- عنقود 5 = 0x05 00 00 00 (بداية الدليل الجذر)
- في هذا المثال: 256 قطاع = عنقود واحد (256 قطاع = 1 عنقود)
- العنقود 2 يبدأ في القطاع 18,432 (0x00 48 00 00) (بداية Bitmap)
- العنقود 5 = 0x05 (المحدد أعلاه) = 1280 قطاع (256 قطاع × 5 عنقود) + 18,432 (موقع بداية منطقة البيانات)— 2 عنقود (512 قطاع) = القطاع 19,200
- إذن، عنقود بداية الدليل الجذر عند العنقود 5 وموقعه القطاع 19,200

37. ^ بعض الوثائق تضع ActiveFat في حقل منفصل عند 0x6B بطول 1 بايت أي بعد أعلام وحدة التخزين، ولا أعرف إن كان هذا تطبيق خاص من exFAT أو خطأ من الكاتب!

38. ^ العدد الإجمالي لوحدات التخصيص على.وحدة التخزين Allocation Units، وتسمى أيضا Bit Count وهي عدد Bits المستخدمة في تعقب توزيع العناقيد Allocation العنقود هو وحدة التخصيص Allocation Unit في exFAT، و 2^{32}-11 هو أقصى عدد للعناقيد يمكن وصفه (أو بالضبط (0xFFFFFFF5 + 1)) و 2^{25} -1 هو أقصى حجم للعنقود (32 ميغابايت).

نظريا، حجم وحدة التخزين الأقصى قد يصل إلى 4,294,967,285 عنقود (2^{32} − 11) مع 33,554,431 بايت لكل عنقود (2^{25} − 1) = حوالي 128 بيتابايت.

حاليا الحجم مقيد كذلك بمخطط العنونة LBA48، كما هو الحال مع حجم قطاع 512، فقط 2^{48} × 512 = 2^{57} = 128 يمكن عنونتها.

مثال 1,007,872 وحدة (كل وحدة يمثلها بت واحد) = 0x00 61 0F 00

39. ^ حاليا نظام ملفات exFAT، يستخدم ثلاثة مجاميع للتدقيق المجموع لضمان سلامة أو تكامل البيانات الوصفية:

- قطاع تدقيق مجموع VBR: للتحقق من سلامة بيانات 11 قطاع السابقة في VBR، (بدون حساب حقول 3 بايت؛ الأعلام والنسبة المئوية).
- تدقيق مجموع upcase table: وهو جدول ثابت ولا يجب أن يتغير. أي تلف فيه، قد يمنع تحديد مواقع الملفات؛ ويستخدم للتحويل أسماء الملفات إلى محارف كبيرة عند البحث عن موقع الملف.
- تدقيق المجموع من أجل مجموعات ملفات الأدلة File set.: عدة تسجيلات دليل تستخدم في تحديد الملف وتسمى file set. وتملك بيانات وصفية تشمل: اسم الملف، أختام زمنية، خصائص، عنوان أول موقع عنقود بيانات، طول الملف.

40. ^ مدخلات الدليل الجذر في exFAT

اسم مدخلة الدليل		شفرة	مدخلة ضرورية	مدخلة أولية	نوع المدخلة
مصفوفة توزيع العناقيد	Allocation Bitmap	1	*	*	0x81
جدول المحارف الكبيرة (من أجل تحويل المحارف)	Up-case Table	2	*	*	0x82
لصيقة وحدة التخزين (اسم القسم)	Volume Label	3	*	*	0x83
ملف	File	5		*	0x85
معرف وحدة التخزين (GUID)	Volume GUID	0	*	*	0xA0
حشوة TexFAT	TexFAT Padding	1		*	0xA1
قائمة التحكم بالنفاذ ACL (أذون نظام الملفات)، في نظام Win CE	Windows CE Access Control Table	2		*	0xA2
(معلومات توزيع الملفات !)	Stream Extension	0	*		0xC0
اسم ملف	File Name	1	*		0xC1

41. ^ مدخلة مصفوفة توزيع العناقيد Allocation Bitmap Directory Entry

وصف	حجم	إزاحة	رمز تذكري
نوع المدخلة 0x81	1	(0x00 (0	EntryType
أعلام تدل على مصفوفة العناقيد Allocation Bitmap التي تصفها المدخلة المحددة أعلام مصفوفة Bitmap: حجم 1، بتات 0، رمز تذكري BitmapIdentifier: 0 = الأولى، 1 = الثانية أعلام مصفوفة Bitmap: حجم 7، بتات 1، رمز تذكري Reserved: محجوزة	1	(0x01 (1	BitmapFlags
محجوزة	18	(0x02 (2	Reserved
أول عنقود	4	(0x14 (20	First Cluster
طول البيانات	8	(0x18 (24	Data Length

- عدد المصفوفات bitmaps ومن ثم عدد مدخلات Bitmap Allocation يساوي عدد جداول FATs. (في بيئة TexFAT تستخدم نسختان من جدول FATs)،
- بت 0 في الأعلام يرجع إلى النسختان النشيطتان FAT + Bitmap.(ستكون أول Allocation Bitmap مع أول FAT، وثاني Allocation Bitmap مع ثاني FAT).
- في قطاع الإقلاع حقل ActiveFat يحدد النسختان النشيطتان. حجم Bitmap بـ بايتات يجب أن يكون عدد العناقيد في وحدة التخزين مقسوم على 8 مع حساب الكسور وتدوير النتيجة.

42. ^ مقارنة بين exFAT و FAT32

الميزّة	FAT32	exFAT
حجم وحدة التخزين الأقصى	8 تيرابايت [1]	128 بيتابايت
حجم الملف الأقصى	4 جيجابايت	16 إكسابايت
حجم العنقود الأقصى	32 كيلوبايت [2]	32 ميغابايت
عدد العناقيد الأقصى	2^{28}	2^{32}
طول اسم الملف الأقصى	255	255
الفصل الزمني أو دقة تاريخ/وقت Date/Time resolution	2 ثانية	10 ملى ثانية
معرف نوع القسم في MBR	0x0B, 0x0C	0x07

[1] ويندوز لا يستطيع تهيئة وحدة التخزين FAT32 الأكبر من 32 جيجابايت، رغم ذلك، يدعم وحدة التخزين الأكبر التي تنشأ عن تطبيقات الطرف الثالث، و 16 تيرابايت هو أقصى حجم لوحدة التخزين المهيئة باستخدام عنقود 64 كيلوبايت.

[2] وفقا لـ KB184006 من مايكروسوفت، العناقيد لا يمكن أن تكون بحجم 64 كيلوبايت أو أكبر، مع ذلك، بعض تطبيقات الطرف الثالث تدعم الحجم الذي يصل إلى 64 كيلوبايت.

مراجع

1. ^ أ ب ت ث، كومباك؛ فينكس؛ إنتيل (1996-01-11). ملف PDF "مواصفة إقلاع نظام BIOS رقم 1.01".
2. ^ أ ب، كومباك؛ فينكس؛ إنتيل (1994-05-05). ملف PDF "مواصفة نظام BIOS الذي يدعم معيار "القرص والتشغيل - رقم 1.0A".
3. ^ روبرت إليوت (2010-01-04). "ملحق شفرة إقلاع MBR الهجين مع مواصفة الأقراص، نسخة EDD-4". موقع لجنة المعايير t13.org.
4. ^ موقع "*The PC Guide*". موضوع " قطاعات إقلاع القسم" (وحدة التخزين).
5. ^ مدونة hakzone

لمعلومات أكثر عن تلك الأنظمة راجع لوائح الموسوعة الحرة (الانجليزية)

أنظمة دوس	إصدارات مايكروسوفت ويندوز
أنظمة الملفات	أنظمة التشغيل
محملات الاقلاع	أنظمة دوس مرتبة زمنيا

تَنْبِيهٌ

لا توجد أية مصادر عربية في هذه الكتيبات ! باستثناء بعض المصطلحات القليلة من قاموس عرب آنز.

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

جهاد

فبراير/شباط 2016

يناير/كانون الثّاني 2019

تمت بحمد الله