تمثيل البيانات داخل الحاسب الآلي:

يستخدم الحاسب نظم التشفير (Coding Systems) لتمثيل الرموز المختلفة الموجودة على لوحة المفاتيح بطريقة ثنائية ومن هذه النظم ما يلي:

١) نظام BCD :

وهي اختصار لـ Binary Coded Decimal والتي تعني نظام الأرقام العشرية الممثلة ثنائياً. يمثل هذا النظام الرموز بواسطة 6 خانات (6 Bits) .

يوضح الجدول التالي طريقة تمثيل البيانات بنظامBCD :

منطقة الدليل	التمثيل الرقمي للرمز									
	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001
00	0	1	2	3	4	5	6	7	8	9
11		A	B	C	D	E	F	G	H	I
10		J	K	L	M	N	O	P	Q	R
01			S	T	U	V	W	X	Y	Z

يتكون هذا النظام فقط من ٦ خانات مما يعني أن عدد رموز هذا النظام هي(2^6)أي 64 رمز فقط.

مثال تمثيل KFU بنظام BCD يكون كالآتي:

100010110110010100

٢) نظام EBCDIC :

وهي اختصار لـ Extended Binary Coded Decimal Interchange والتي تعني الشفرة الموسعة للنظام العشري الممثل ثنائياً ويستخدم هذا النظام 8 خانات لتمثيل الرموز المختلفة الموجودة على لوحة المفاتيح كما مبين بالجدول التالي:

منطقة الدليل	التمثيل الرقمي للرمز									
	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001
1111	0	1	2	3	4	5	6	7	8	9
1100		A	B	C	D	E	F	G	H	I
1101		J	K	L	M	N	O	P	Q	R
1110			S	T	U	V	W	X	Y	Z

يستخدم هذا النظام 8 خانات لتمثيل الرموز مما يعني 2^8 احتمال أي 256 رمز .

مثال: مثلي كلمة KFU بنظام EBCDIC .

— 11010010110001101110100

٣) نظام ASCII:

وهي اختصار لـ American Standard Code for Information Interchange والتي تعني الشفرة الأمريكية القياسية لتبادل المعلومات.

ويوجد نظامان لآسكي أحدهما يستخدم 7 خانات لتمثيل الرموز الموجودة على لوحة المفاتيح بـ 2^7 =128 رمز. والآخر يستخدم8 خانات وبالتالي يمكن تمثيل 2^8 أي 256حرف.

يوضح الجدول التالي طريقة تمثيل البيانات بنظام(ASCII (7-Bit :

الدليل	التمثيل الرقمي للرمز															
	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
011	0	1	2	3	4	5	6	7	8	9						
100		A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
101	P	Q	R	S	T	U	V	W	X	Y	Z					

مثال: مثلي KFU بنظام (ASCII (7-Bit.

1001011100011010101

ويوضح الجدول التالي طريقة تمثيل البيانات بنظام(ASCII (8-Bit :

الدليل	التمثيل الرقمي للرمز															
	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
0101	0	1	2	3	4	5	6	7	8	9						
1010		A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1011	P	Q	R	S	T	U	V	W	X	Y	Z					

فيصبح تمثيل KFU بنظام (ASCII (8-Bit كما لآتي:

1010101110100110110101

تمرين:

مثلي كلمة COMPUTER

(أ) بنظام **BCD**.

(ب) بنظام **EBCDIC**.

(ج) بنظام ASCII (7-Bit).

(د) بنظام ASCII (8-Bit).

بت التدقيق Parity Bit (check digit) :

هي بت إضافية تلحق بالبايت المنقول من المعلومات وتستخدم في الكشف عن الأخطاء عند نقل المعلومات.

فالحاسب الذي يستخدم شفرة مكونة من 8 Bits مثل نظام EBCDIC سوف تكون هنالك بت تاسعة إضافية لغرض التدقيق.

إحدى طرق التأكد من أن المعلومات نقلت بصورة صحيحة هي أن يتم إرسالها مرتين والمقارنة بين الإرسالين، لكن هذه الطريقة قليلة الكفاءة لأنها تضاعف الوقت والتكلفة.

لذا تم استخدام بت التدقيق كحل بديل لتلك الطريقة.

هنالك حواسيب يكون فيها عدد خانات الرقم 1 في كل بايت مرسل عدد زوجي وتسمى حواسيب زوجية التدقيق، بينما الحواسيب التي يكون فيها عدد خانات الرقم واحد عدداً فردياً تسمى حواسيب فردية التدقيق.

في الحواسيب زوجية التدقيق إذا كان عدد خانات الرقم 1 عدداً فردياً في البايت المرسل، فان بت التدقيق يجب أن تكون1. وذلك لضمان أن يكون عدد خانات الرقم 1 عدد زوجي.

وإذا كان عدد خانات الرقم 1 زوجياً فان بت التدقيق المضافة يجب أن تكون صفر تلقائياً.

مثلاً: تمثيل الرقمين 2 و 3 على حاسب زوجي التدقيق باستخدام نظام EBCDIC هو

2 تمثيلها:

11110010	1

← بت التدقيق (وقيمتها 1 لان عدد خانات 1 عدد فردي)

3 تمثيلها:

11110011	0

← بت التدقيق (وقيمتها 0 لان عدد خانات 1 عدد زوجي)

عندما يتم إرسال بيانات حاسب زوجي التدقيق فان الحاسب المستقبل يفترض أن عدد خانات الرقم 1 المستقبل في كل بايت يجب أن يكون زوجي فإذا أكتشف الحاسب أن هنالك بايت به عدد فردي فانه يطلب تلقائياً إعادة الإرسال Retransmission .

أما الحواسيب فردية التدقيق فإنها تعمل بنفس المفهوم باستثناء أنها تستخدم بت التدقيق للتأكد من أن عدد خانات الرقم 1 المرسلة يجب أن يكون فردياً وإلا فانه يطلب إعادة الإرسال.

جدول تخصيص الملفات (File Allocation Table(FAT)) :

يستخدم العدد الثنائي ذو الـ 16 بت (خانة) للإشارة إلى عدد الوحدات التخزينية التي تخزن على القرص الصلب حيث تسع كل وحدة تخزينية 512 Bytes (½ KB) وهذه الوحدات تسمى القطاعات (Sectors) وبما أن العدد الثنائي 2^{16} = 65536 لذا يشير FAT 16 إلى 65536 عنوان (Address) للوحدات التخزينية أي ما يقارب

512 B * 65536 = 33554432 Byte = 32768 KB = 32 MB

وهذا يعني أن الحجم الأقصى للقرص الصلب الذي يستطيع جدول تخصيص الملفات FAT 16 التعامل معه هو 32 MB . وكان هذا هو الحد الأقصى لأول قرص صلب ولكن مع التطور وتزايد أحجام البرامج كان لابد من استخدام أحجام أكبر للقرص الصلب فكان لابد من وجود طريقة لجدولة مواقع الملفات على هذا القرص، لذلك تم استخدام تعبير جديد هو العنقود (Cluster) والذي هو عبارة عن مجموعة من القطاعات (Sectors).

على القرص الواحد حدد لكل عنقود ثمانية قطاعات فأصبح حجم العنقود 4 KB (8x ½ KB)

وهذا يعني أننا باستخدام FAT 16 يمكننا التجوال ضمن مساحة تخزينية تساوي :

4 KB x 65536 =262144 KB =256 MB

ولكن التطور في الأحجام الضخمة للبرامج لم يتوقف وكان لابد من السعي من جديد للتغلب على هذا الموضوع ، لذلك خصص لكل عنقود عدد أكبر من القطاعات وصل إلى 64 قطاع وبالتالي أصبح حجم العنقود :

64 x 512 B = 32768 B = 32 KB

وهذا ذاد من الحجم الأقصى للقرص الصلب فأصبح:

65536 x 32 KB =2097152 KB = 2048 MB = 2 GB

وهذا هو الحد الأقصى للقرص الصلب الذي نستطيع التعامل معه بنظام Dos ونظام Windows95 . وبالتالي يجب تجزئة الأقراص الصلبة ذات الأحجام أكبر من 2GB إلى عدة أجزاء باستخدام الأمر FDisk وبحد أقصى للكل جزء لا يزيد عن 2 GB .

أصبح الآن بإمكاننا تخصيص عدد أكبر من القطاعات لكل عنقود وبالتالي زيادة حجم القرص الصلب (أو جزء منه) وبمجال عناوين لا يزيد عن 65536 (2^{16}) وهذا ربح جيد.

إذا كانت لدينا ملفات صغيرة الحجم مثل بعض ملفات نظام الويندوز التي لا تزيد عن 128 byte (حرف) وحيث أننا لا نستطيع تخزين أكثر من ملف واحد في العنقود الواحد حتى ولو كان حجم هذه الملفات صغيراً جداً لأنه لا يمكن الإشارة إلى عنقود (cluster) واحد من خلال عنوانين في جدول تخصيص الملفات وبالتالي سيتم حجز أماكن إضافية فائضة لمثل هذه الملفات صغيرة الحجم .

لذلك لجأت شركة مايكروسوفت جدول مواقع الملفات FAT 32 مع نظام Windows 98 وما بعده مع المحافظة على حجم 4KB لكل عنقود وبهذه الطريقة نستطيع باستخدام FAT 32 تخصيص حجم هائل لكل جزء من أجزاء القرص الصلب يساوي:

$2^{32} \times 4\ KB = 16\ TB$

الوسائط المغناطيسية Magnetic Media :

الشريط المغناطيسي Magnetic Tape : هو من الوسائط الهامة لعمل النسخ الإحتياطي Back up .

١ كثافة التسجيل (Recording Density) (RD):

تقاس كثافة التسجيل بعدد البايت التي يمكن تسجيلها في البوصة الواحدة وتميز بـ BPI وتعني عدد البايتات في البوصة(Bytes per Inch) .

٢ الفراغ بين السجلات (Inter record gap) :

وهو الفراغ بين سجل وآخر والناتج عن وجوب أن تكون سرعة مرور الشريط ثابتة.

٣ الفراغ بين التجميعات (Inter block gap) :

وهو الفراغ بين تجميعة وآخري والناتج عن وجوب أن تكون سرعة مرور الشريط ثابتة.

٤ المعامل التجميعي (Blocking Factor) (BF):

عدد السجلات الموجودة في التجميعة الواحدة.

مثال ١ :

شريط مغناطيسي طوله 2400 قدم (القدم = 12 بوصة) وكثافة تسجيله 1600 BPI (عدد البايتات أو الحروف في البوصة) وطول الفراغات 0.4 بوصة. أستخدم هذا الشريط لتخزين ملف يتكون من عدد من السجلات طول السجل الواحد = 80 Byte .

أوجدي عدد السجلات التي يمكن تخزينها في هذا الشريط علماً بأن BF = 1 .

الحل:

طول الشريط اللازم لتخزين سجل واحد (الجزء البياني) = طول السجل بالبايت

كثافة التسجيل(RD)

= 80/1600 =0.05 Inch

طول الشريط اللازم لتخزين السجل (بياني +فراغ)=

0.05+0.4 = 0.45 Inch

عدد السجلات التي يمكن تخزينها في الشريط = طول الشريط الكلي

طول الشريط للسجل الواحد

= 12x2400/0.45 = 64000 سجل

مثال ٢:

مستخدمة المثال السابق أوجدي عدد السجلات التي يمكن تخزينها في الشريط إذا كان BF=8 .

الحل:

طول الشريط اللازم لتخزين سجل واحد (الجزء البياني) = طول السجل بالبايت

كثافة التسجيل(RD)

= 80/1600 =0.05 Inch

طول الشريط اللازم لتخزين التجميعة (بياني)= الطول البياني للسجل x BF

= 0.05 x 8 =0.4 بوصة

الشريط اللازم لتخزين التجميعة (بياني+ فراغ)= 0.4 + 0.4 = 0.8 بوصة

عدد التجميعات التي يمكن تخزينها في الشريط = طول الشريط الكلي

طول الشريط للتجميعة(بياني وفراغ)

= 12x2400/0.8 = 36000 تجميعة

عدد السجلات التي يمكن تخزينها في الشريط = عدد التجميعات x BF =

8 x 36000= 288000 سجل

مثال ٣:

احتجنا ٤ أشرطة مغناطيسية لتخزين ملف طول سجله 500 Byte وكثافة التسجيل فيه 12500 BPI والفراغ بين السجلات 0.3 بوصة وخزنت السجلات بدون تجميع.

أوجدي معامل التجميع BF المناسب لتخزين الملف في شريط واحد.

الحل:

نفرض المعامل التجميعي BF= X

طول الشريط اللازم لتخزين سجل واحد (الجزء البياني) = طول السجل بالبايت

كثافة التسجيل(RD)

= 500/12500 * 4 = 0.04 بوصة

طول الشريط اللازم لتخزين عدد **X**سجل (دون تجميع)=

$(0.04+0.3)* X= 0.34 X$ ——————— (1)

طول الشريط اللازم لتخزين عدد **X**سجل (بالتجميع)=

$0.3+0.04 X$ ——————— (2)

بما أن المعادلة الثانية =¼ المعادلة الأولى:

$¼(0.34X)= (0.3+ 0.04X)$

$0.34X=1.2 + 0.16 X$

$0.18X=1.2$

$X=1.2/0.18 = 6.666666666666667 ≈ 7$