LBA to CHS Translation

CHS translation is needed only for disk drives between 528MB and 8GB. Below 528MB CHS addresses do not require translation. Above 8GB CHS addressing is not supported by ATA hard disk drives or by system BIOS software. Most systems are finally making the switch to using LBA addressing on all capacity drives.

Introduction

This is very technical. Please read carefully. There is lots of information here that can sound confusing the first time you read it.

Why is an understanding of how a BIOS works so important? The basic reason is that the information returned by INT 13H AH=08H is used by FDISK, it is used in the partition table entries within a partition record (like the Master Boot Record) that are created by FDISK, and it is used by the small boot program that FDISK places into the Master Boot Record. The information returned by INT 13H AH=08H is in cylinder/head/sector (CHS) format, it is not in LBA format. The boot processing done by your computer's BIOS (INT 19H and INT 13H) is all CHS based.

Read this so that you are not confused by all the false information going around that says "LBA solves the >528MB problem".

Read this so that you understand the possible data integrity problem that a WD EIDE type BIOS creates. Any BIOS that has a "LBA mode" in the BIOS setup could be a WD EIDE BIOS. Be very careful and NEVER chage the "LBA mode" setting after you have partitioned and installed your software.

Definitions

- 528MB - The maximun drive capacity that is supported by 1024 cylinders, 16 heads and 63 sectors (1024x16x63x512). This is the limit for CHS addressing in the original IBM PC/XT and IBM PC/AT INT 13H BIOS.
- 8GB - The maximum drive capacity that can be supported by 1024 cylinders, 256 heads and 63 sectors (1024x256x63x512). This is the limit for the BIOS INT 13H AH=0xH calls.
- ATA - AT Attachment. The real name of what is widely known as IDE.
- CE Cylinder - Customer Engineering cylinder. This is the last cylinder in P-CHS mode. IBM has always reserved this cylinder for use of disk diagnostic programs. Many BIOS do not account for it correctly. It is of questionable value these days and probably should be considered obsolete. However, since there is no industry wide agreement, beware. There is no CE Cylinder reserved in the L-CHS address. Also beware of diagnostic programs that do not realize they are operating in L-CHS mode and think that the last L-CHS cylinder is the CE Cylinder.
- CHS - Cylinder/Head/Sector. This is the traditional way to address sectors on a disk. There are at least two types of CHS addressing: the CHS that is used at the INT 13H interface and the CHS that is used at the ATA device interface. In the MFM/RLL/ESDI and early ATA days the

CHS used at the INT 13H interface was the same as the CHS used at the device interface.

Today we have CHS translating BIOS types that can use one CHS at the INT 13H interface and a different CHS at the device interface. These two types of CHS will be called the logical CHS or L-CHS and the physical CHS or P-CHS in this document. L-CHS is the CHS used at the INT 13H interface and P-CHS is the CHS used at the device interface.

The L-CHS used at the INT 13 interface allows up to 256 heads, up to 1024 cylinders and up to 63 sectors. This allows support of up to 8GB drives. This scheme started with either ESDI or SCSI adapters many years ago.

The P-CHS used at the device interface allows up to 16 heads up to 65535 cylinders, and up to 63 sectors. This allows access to 2^28 sectors (136GB) on an ATA device. When a P-CHS is used at the INT 13H interface it is limited to 1024 cylinders, 16 heads and 63 sectors. This is where the old 528MB limit originated.

ATA devices may also support LBA at the device interface. LBA allows access to approximately 2^28 sectors (137GB) on an ATA device.

A SCSI host adapter can convert a L-CHS directly to an LBA used in the SCSI read/write commands. On a PC today, SCSI is also limited to 8GB when CHS addressing is used at the INT 13H interface.

- EDPT - Enhanced fixed Disk Parameter Table. This table returns additional information for BIOS drive numbers 80H and 81H. The EDPT for BIOS drive 80H is pointed to by INT 41H. The EDPT for BIOS drive 81H is pointed to by INT 46H. The EDPT is a fixed disk parameter table with an AxH signature byte. This table format returns two sets of CHS information. One set is the L-CHS and is probably the same as returned by INT 13H AH=08H. The other set is the P-CHS used at the drive interface. This type of table allows drives with >1024 cylinders or drives >528MB to be supported. The translated CHS will have <=1024 cylinders and (probably) >16 heads. The CHS used at the drive interface will have >1024 cylinders and <=16 heads. It is unclear how the IBM defined CE cylinder is accounted for in such a table. Compaq probably gets the credit for the original definition of this type of table.
- FDPT - Fixed Disk Parameter Table - This table returns additional information for BIOS drive numbers 80H and 81H. The FDPT for BIOS drive 80H is pointed to by INT 41H. The FDPT for BIOS drive 81H is pointed to by INT 46H. A FDPT does not have a AxH signature byte. This table format returns a single set of CHS information. The L-CHS information returned by this table is probably the same as the P-CHS and is also probably the same as the L-CHS returned by INT 13H AH=08H. However, not all BIOS properly account for the IBM defined CE cylinder and this can cause a one or two cylinder difference between the number of cylinders returned in the AH=08H data and the FDPT data. IBM gets the credit for the original definition of this type of table.
- LBA - Logical Block Address. Another way of addressing sectors that uses a simple numbering scheme starting with zero as the address of the first sector on a device. The ATA standard requires that cylinder 0, head 0, sector 1 address the same sector as addressed by LBA 0. LBA addressing can be used at the ATA interface if the ATA device supports it. LBA addressing is also used at the INT 13H interface by the AH=4xH read/write calls.
- L-CHS, Logical CHS. The CHS used at the INT 13H interface by the AH=0xH calls. See CHS above.
- MBR - Master Boot Record (also known as a partition table) - The sector located at cylinder 0 head 0 sector 1 (or LBA 0). This sector is created by an "FDISK" utility program. The MBR may be the only partition table sector or the MBR can be the first of multiple partition table sectors that form a linked list. A partition table entry can describe the starting and ending sector addresses of a partition (also known as a logical volume or a logical drive) in both L-CHS and LBA form. Partition table entries use the L-CHS returned by INT 13H AH=08H. Older FDISK programs may not compute valid LBA values.
- OS - Operating System.
- P-CHS, Physical CHS. The CHS used at the ATA device interface. This CHS is also used at the INT 13H interface by older BIOS's that do not support >1024 cylinders or >528MB. See CHS above.

Background and Assumptions

First, please note that this is written with the OS implementor in mind and that I am talking about the possible BIOS types as seen by an OS during its hardware configuration search.

It is very important that you not be confused by all the misinformation going around these days. All OS's that want to be co-resident with another OS (and that is all of the PC based OS's that I know of)

MUST use INT 13H to determine the capacity of a hard disk. And that capacity information MUST be determined in L-CHS mode. Why is this? Because: 1) FDISK and the partition tables are really L-CHS based, and 2) MS/PC DOS uses INT 13H AH=02H and AH=03H to read and write the disk and these BIOS calls are L-CHS based. The boot processing done by the BIOS is all L-CHS based. During the boot processing, all of the disk read accesses are done in L-CHS mode via INT 13H and this includes loading the first of the OS's kernel code or boot manager's code.

Second, because there can be multiple BIOS types in any one system, each drive may be under the control of a different type of BIOS. For example, drive 80H (the first hard drive) could be controlled by the original system BIOS, drive 81H (the second drive) could be controlled by a option ROM BIOS and drive 82H (the third drive) could be controlled by a software driver. Also, be aware that each drive could be a different type, for example, drive 80H could be an MFM drive, drive 81H could be an ATA drive, drive 82H could be a SCSI drive.

Third, not all OS's understand or use BIOS drive numbers greater than 81H. Even if there is INT 13H support for drives 82H or greater, the OS may not use that support.

Fourth, the BIOS INT 13H configuration calls are:

- AH=08H, Get Drive Parameters. This call is restricted to drives up to 528MB without CHS translation and to drives up to 8GB with CHS translation. For older BIOS with no support for >1024 cylinders or >528MB, this call returns the same CHS as is used at the ATA interface (the P-CHS). For newer BIOS's that do support >1024 cylinders or >528MB, this call returns a translated CHS (the L-CHS). The CHS returned by this call is used by FDISK to build partition records.
- AH=41H, Get BIOS Extensions Support. This call is used to determine if the IBM/Microsoft Extensions or if the Phoenix Enhanced INT 13H calls are supported for the BIOS drive number.
- AH=48H, Extended Get Drive Parameters. This call is used to determine the CHS geometries, LBA information and other data about the BIOS drive number.
- The FDPT or EDPT. While not actually a call, but instead a data area, the FDPT or EDPT can return additional information about a drive.
- Other tables. The IBM/Microsoft extensions provide a pointer to a drive parameter table via INT 13H AH=48H. The Phoenix enhancement provides a pointer to a drive parameter table extension via INT 13H AH=48H. These tables are NOT the same as the FDPT or EDPT.

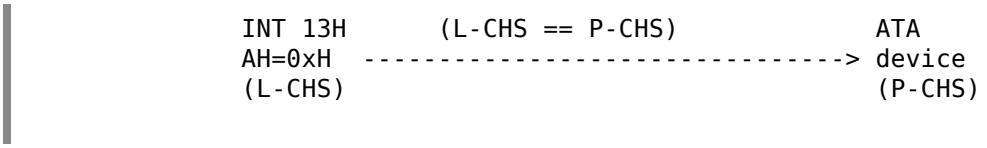
Note: The INT 13H AH=4xH calls duplicate the older AH=0xH calls but use a different parameter passing structure. This new structure allows support of drives with up to 2^64 sectors (really BIG drives). While at the INT 13H interface the AH=4xH calls are LBA based, these calls do NOT require that the drive support LBA addressing.

CHS Translation Algorithms

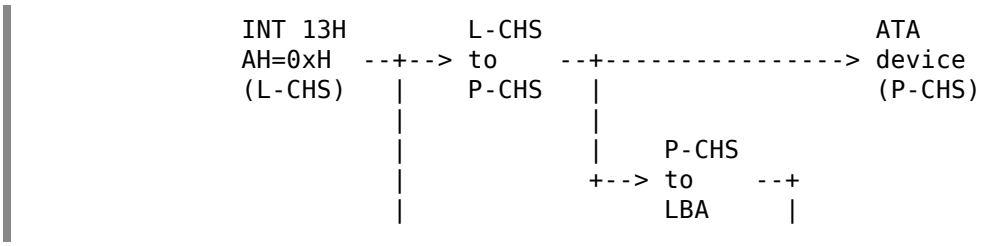
As you read this, do not forget that all of the boot processing done by the system BIOS via INT 19H and INT 13H use only the INT 13H AH=0xH calls and that all of this processing is done in CHS mode.

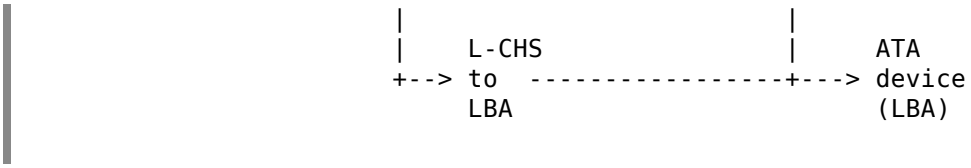
First, lets review all the different ways a BIOS can be called to perform read/write operations and the conversions that a BIOS must support.

- An old BIOS (like BIOS type 1 below) does no CHS translation and does not use LBA. It only supports the AH=0xH calls:

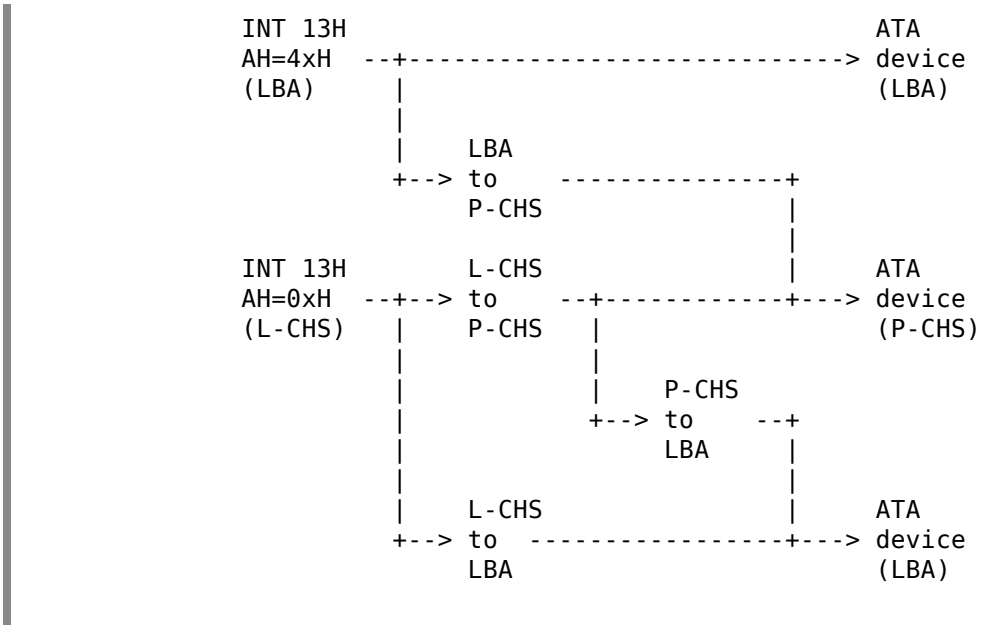


- A newer BIOS may support CHS translation and it may support LBA at the ATA interface:





- A really new BIOS may also support the AH=4xH in addtion to the older AH\0xH calls. This BIOS must support all possible combinations of CHS and LBA at both the INT 13H and ATA interfaces:



You would think there is only one L-CHS to P-CHS translation algorithm, only one L-CHS to LBA translation algorithm and only one P-CHS to LBA translation algorithm. But this is not so. Why? Because there is no document that standardizes such an algorithm. You can not rely on all BIOS's and OS's to do these translations the same way.

The following explains what is widely accepted as the "correct" algorithms.

An ATA disk must implement both CHS and LBA addressing and must at any given time support only one P-CHS at the device interface. And, the drive must maintain a strick relationship between the sector addressing in CHS mode and LBA mode. Quoting the ATA-2 document:

LBA = ((cylinder * heads_per_cylinder + heads) * sectors_per_track) + sector - 1

This algorithm can be reversed such that an LBA can be converted to a CHS:

```
cylinder = LBA / (heads_per_cylinder * sectors_per_track)
temp = LBA % (heads_per_cylinder * sectors_per_track)
head = temp / sectors_per_track
sector = temp % sectors_per_track + 1
```

Back