***Сервисы BIOS*  Int 1Ah** позволяют считывать и модифицировать значения *системного таймера, даты и времени* , а также установки *будильника* часов реального времени CMOS RTC. Перечисленные ниже номера функций указываются при вызове в регистре АН.

1. АН=0 — чтение системного таймера (двойного слова по адресу 40:006Eh в BIOS Data Area, инкрементируемого по прерываниям от канала 0 счетчика- таймера 8253/8254 примерно раз в 55 мс. Таймер обнуляется при выполнении теста POST после аппаратного сброса). Возвращает значение таймера, в СХ — старшую часть, в DX — младшую. AL=0, если за последние 24 часа не было переполнения таймера. В современных версиях сброс AL возвращает счетчик переполнений таймера, хранящийся в ячейке 40:0070h (в старых версиях это был флаг).

2. АН=1 — установка системного таймера (СХ — старшая часть, в DX — младшая) и сброс флага (счетчика) переполнения таймера в ячейке 40:0070h. В случае ошибки устанавливается флаг CF=1.

3. АН=2 — чтение времени из RTC. Возвращает в упакованном BCD-формате час (в регистре СН), минуту (CL), секунду (DH) и признак коррекции летнего/зимнего времени (DL=1 — коррекция используется, DL=0 — нет). Признаком успешной операции является флаг CF=0. Во избежание ошибок некоторых BIOS при вызове флаг CF должен быть сброшен.

4. АН=3 — установка времени в RTC, назначение регистров и признак результата аналогичен функции 2.

5. АН=4 — чтение даты из RTC. Возвращает в упакованном BCD-формате век (в регистре СН), две старшие цифры года (CL), месяц (DH) и день (DL). Признаком успешной операции является флаг CF=0. Во избежание ошибок некоторых BIOS при вызове флаг CF должен быть сброшен.

6. АН=5 — установка даты в RTC, назначение регистров и признак результата аналогичен функции 4.

7. АН=6 — установка времени срабатывания будильника RTC. Возвращает в упакованном BCD-формате час (в регистре СН), минуту (CL) и секунду (DH). Если будильник уже установлен, переустановка не производится и возвращается флаг CF=1. При срабатывании будильник вызывает прерывание Int 4Ah.

8. АН=7 — отмена установки будильника.

***Сервисы BIOS*  Int 15h** позволяют программировать *таймер CMOS RTC* — вводить задержку или запускать таймер установки флага, указывая время в микросекундах (СХ — старшее слово, DX — младшее). Нулевое значение интервала не вызывает никаких действий. Достижимое разрешение в зависимости от производительности ПК может достигать единиц миллисекунд, максимальная выдержка — около 70 часов. Перечисленные ниже номера функций указываются при вызове в регистре АН или АХ.

9. AH=86h — задержка на заданное время. Управление будет возвращено вызвавшему процессу только через указанный интервал. По окончании задержки будет установлен бит 7 в ячейке BDA 0040:00A0. Таймер может оказаться занятым, тогда вызов сразу возвратит флаг CF=1 (при успехе CF=0, а в AL окажется маска, записанная в 8259А#2).

10. AX=8300h — запуск таймера, устанавливающего флаг после указанной задержке бит 7 в ячейке, заданной регистрами ES:BX. При успешном запуске CF=0; если таймер занят (он один) — CF=1 и AL=0. Управление возвращается процессу сразу, а флаг будет установлен через заданное время. Перед завершением программа, запускавшая таймер, должна его сбросить функцией 8301h (во-первых, чтобы освободить; во-вторых, чтобы снять «адскую машинку», которая неожиданно сама может изменить значение ячейки памяти, вполне возможно уже задействованную другим, ничего не «подозревающим» процессом).

11. AX=8301h — сброс того же таймера.

***Прерывание 16h***

**Функция 00h (10h).**Чтение символа клавиатуры с ожиданием.

Читает из кольцевого буфера ввода символ и скан-код. После считывания они удаляются из буфера и возвращаются в регистре *AX*. *Если буфер пуст, ожидает ввода.*Каждой клавише на клавиатуре соответствует так называемый скан-код, соответствующий только этой клавише. Этот код посылается клавиатурой при каждом нажатии и отпускании клавиши и обрабатывается в BIOS обработчиком прерывания *Int 09h*. Функция *00h*даёт возможность получить код нажатия, не перехватывая этот обработчик. Если нажатой клавише соответствует *ASCII-*символ, то:

*AL – ASCII*-код символа, *AH –*скан-код клавиши.

Если нажатой клавише соответствует расширенный *ASCII-*код, то:

+

*AL – 00h, AH – расширенный ASCII-код. Вызов: AH = 00h (83/84-key).*

Возврат: *AL = ASCII*-код символа, изображённый на клавише/*00h*, *AH*= скан-код/расширенный *ASCII*-код клавиши.

**Функция 10h**– усовершенствованный вариант функции 00h для расширенной клавиатуры (101/102-*key*). Позволяет получить расширенные *ASCII-*коды для клавиш *F11, F12*, а также для ряда других комбинаций. В качества признака управляющих клавиш или их комбинаций, помимо значения *00h*, используются *0Ah, 0Dh*и *E0h*.

**Функция 01h (11h).**Поверка буфера клавиатуры на наличие в нём символа. Определяет, имеются ли в кольцевом буфере ожидающие ввода символы; возвращает

флаг ожидания и сам символ при его наличии. Однако символ и его скан-код не извлекаются из буфера и могут быть снова получены при повторном вызове функции *00h Int 16h*. Данная функция относится к числу асинхронных: определив состояние буфера ввода, она возвращает управление программе.

Вызов: *AH = 01h*(83/84-*key*), *11h*(101/102*-key*).

Возврат: *ZF =*1, если буфер пуст и *ZF*= 0, если в буфере имеется ожидающий считывания символ. В этом случае:

*AL = ASCII*-код символа/*00h*, *AH*= скан-код клавиши/расширенный *ASCII-*код. **Функция 11h***–*усовершенствованный вариант функции *01h*для расширенной

клавиатуры (101/102*-key*). Позволяет получить расширенные *ASCII*-коды для клавиш *F11, F12*, а также для ряда других комбинаций. В качестве признака управляющих клавиш или их комбинаций, помимо значения *00h*, используются *0Ah, 0Dh*и *E0h*.

**Функция 02h (12h).**Получение флагов клавиатуры.

Возвращает байт флагов клавиатуры, описывающих состояние управляющих клавиш, записанное в байте (слове) области данных BIOS по адресу *0000h:0417h.*

*Вызов: AH = 02h*

Возврат: *A L*=1-ый байт флагов клавиатуры. Биты байта имеют следующие значения:

**0:**1 – правая *Shift*нажата

**1:**1 – левая *Shift*нажата

**2:**1 – *Ctrl*(любая) нажата

**3:**1 – *Alt*(любая) нажата

***4:****1 – режим Scroll Lock*

**5:**1 – режим *Num Lock*

**6:**1 – режим *Caps Lock*

**7:**1 – режим *Insert*активен

**Функция 12h***–*усовершенствованный вариант функции *02h*для расширенной клавиатуры (101/102-*key*). Выводит такое же значение байта, как и функция *02h*, по адресу

*0000h:0417h*, и, дополнительно, второй байт статуса клавиатуры (адрес *0000h:0418h*) со следующими значениями:

**0:**1 – левая *Ctrl*нажата

**1:**1 – левая *Alt*нажата

**2:**1 – правая *Ctrl*нажата

**3:**1 – правая *Alt*нажата

**4:**1 – нажата Scroll Lock

**5:**1 – нажата Num Lock

**6:**1 – нажата Caps Lock

**7:**1 – нажата SysReg