



This is part of **Family API** which allow to create dual-os version of program runs under OS/2 and DOS

**Note:** This is legacy API call. It is recommended to use 32-bit equivalent

2021/09/17 04:47 · prokushev · [0 Comments](#)

2021/08/20 03:18 · prokushev · [0 Comments](#)

## DosGetDateTime

This call gets the current date and time maintained by the operating system.

### Syntax

DosGetDateTime (DateTime)

### Parameters

;DateTime (PDATETIME) - output : Address of the date and time structure: ::hours (UCHAR) - Current hour ::minutes (UCHAR) - Current minute ::seconds (UCHAR) - Current second ::hundredths (UCHAR) - Current hundredth of a second ::day (UCHAR) - Current day ::month (UCHAR) - Current month ::year (USHORT) - Current year ::timezone (SHORT) - Minutes west of UTC (Universal Time Coordinate) ::weekday (UCHAR) - Current day of the week. Sunday is 0.

### Return Code

rc (USHORT) - return Return code description is: \* 0 NO\_ERROR

### Remarks

The dayofweek value is based on Sunday equal to zero. The value of timezone is the difference in minutes between the current time zone and UTC. This number is positive if it is earlier than UTC and negative if it is later than UTC. For Eastern Standard Time, this value is 300 (5 hours earlier than UTC).

If the application is executing in the OS/2 environment, it is more efficient to obtain these variables by calling DosGetInfoSeg instead of this function. However, applications written to the family API cannot depend on the availability of DosGetInfoSeg.

### Example Code

## C Binding

```
typedef struct _DATETIME { /* date */
    UCHAR    hours;        /* current hour */
    UCHAR    minutes;     /* current minute */
    UCHAR    seconds;     /* current second */
    UCHAR    hundredths;  /* current hundredths of a second */
    UCHAR    day;         /* current day */
    UCHAR    month;       /* current month */
    USHORT   year;        /* current year */
    SHORT    timezone;    /* minutes of time west of UTC */
    UCHAR    weekday;     /* current day of week */
} DATETIME;

#define INCL_DOSDATETIME

USHORT    rc = DosGetDateTime(DateTime);
PDATETIME DateTime; /* Address of date/time structure (returned) */
USHORT    rc; /* return code */
```

This example gets the current time and date.

```
#define INCL_DOSDATETIME

DATETIME DateBuffer;
USHORT    rc;

rc = DosGetDateTime(&DateBuffer); /* Date/Time structure */
```

The following example obtains and prints date and time information. It then changes the system date to 5/10/1987 and prints the updated information.

```
#define INCL_DOSDATETIME
#include <os2.h>

main()
{
    DATETIME    DateTime; /* Structure to hold date/time info. */
    USHORT     rc;

    rc = DosGetDateTime(&DateTime); /* Address of d/t structure */
    printf("Today is %d-%d-%d; the time is %d:%d\n", DateTime.month,
           DateTime.day, DateTime.year, DateTime.hours, DateTime.minutes);
    DateTime.day = 10;
    DateTime.month = 5;
    DateTime.year = 1987;
    printf("The new date is %d-%d-%d; the time is %d:%d\n", DateTime.month,
           DateTime.day, DateTime.year, DateTime.hours, DateTime.minutes);
    rc = DosSetDateTime(&DateTime); /* Address of d/t structure */
```

```
printf("rc is %d\n", rc);  
}
```

## MASM Binding

```
DATETIME struc  
    date_hours      db  ? ;current hour  
    date_minutes    db  ? ;current minute  
    date_seconds    db  ? ;current second  
    date_hundredths db  ? ;current hundredths of a second  
    date_day        db  ? ;current day  
    date_month      db  ? ;current month  
    date_year       dw  ? ;current year  
    date_timezone   dw  ? ;minutes of time west of UTC  
    date_weekday    db  ? ;current day of week  
DATETIME ends  
  
EXTRN  DosGetDateTime:FAR  
INCL_DOSDATETIME EQU 1  
  
PUSH@  OTHER  DateTime      ;Date/time structure (returned)  
CALL   DosGetDateTime
```

Returns WORD

## Note

Text based on [http://www.edm2.com/index.php/DosGetDateTime\\_\(Legacy\)](http://www.edm2.com/index.php/DosGetDateTime_(Legacy))

| Family API |   |  |
|------------|---|--|
| DOS        | Process Manager   | DosBeep DosExit DosSleep DosExecPgm  |
|            | File Manager  | DosChDir DosChgFilePtr DosClose DosDelete DosDupHandle DosMkDir DosMove DosQCurDir DosQCurDisk DosSetFileMode DosOpen DosQFileInfo DosRead DosQFileMode DosQFSInfo DosQVerify DosRmdir DosSelectDisk DosFindClose DosFindFirst DosFindNext DosSetFileInfo DosSetVerify DosWrite DosFileLocks DosSetFHandState DosNewSize DosBufReset DosQFHandState DosSetFSinfo DosShutdown |
|            | Memory Manager  | DosFreeSeg DosSubAlloc DosSubFree DosSubSet DosAllocHuge DosAllocSeg DosReallocHuge DosReallocSeg DosGetHugeShift DosCreateCSAlias   |
|            | NLS   | DosCaseMap DosGetCtryInfo DosGetDBCSEv DosSetCtryCode DosGetCollate DosGetMessage DosInsMessage DosPutMessage  |
|            | Date and Time   | DosSetDateTime DosGetDateTime  |
|            | Devices   | DosDevConfig DosDevIOct1 DosDevIOct2   |
|            | Signals   | DosHoldSignal DosSetSigHandler   |
|            | Misc  | BadDynLink DosGetEnv DosGetMachineMode DosGetVersion DosError DosErrClass DosSetVec  |
| KBD        | KbdCharIn KbdFlushBuffer KbdGetStatus KbdSetStatus KbdStringIn KbdPeek  |  |
| VIO        | VioGetBuf VioGetConfig VioGetCurPos VioGetCurType VioGetPhysBuf VioReadCellStr VioReadCharStr VioScrollUp VioScrollDn VioScrollLf VioScrollRt VioScrUnLock VioSetCurPos VioSetCurType VioSetMode VioGetMode VioShowBuf VioWrtCellStr VioWrtCharStr VioWrtCharStrAtt VioWrtNAttr VioWrtNCell VioWrtNChar VioWrtTTY VioScrLock VioPopUp |  |
| Tools      | BIND  |  |
| Modules    | DOSCALLS.DLL VIOCALLS.DLL KBDCALLS.DLL MSG.DLL  |  |
| Libraries  | API.LIB OS2386.LIB FAPI.LIB DOSCALLS.LIB SUBCALLS.LIB   |  |

2018/08/25 15:05 · prokushev · 0 Comments

From: <https://www.osfree.org/doku/> - **osFree wiki**

Permanent link: <https://www.osfree.org/doku/doku.php?id=en:docs:fapi:dosgetdatetime&rev=1629431080>

Last update: **2021/08/20 03:44**

