

GrdGetTime

Функция(метод) **GrdGetTime** получает время микросхемы таймера ключа.

C

```
int GRD_API GrdGetTime(
    HANDLE hGrd,
    TGrdSystemTime *pGrdSystemTime,
    void *pReserved
);
```

<i>hGrd</i>	хэндл, через который будет выполнена данная операция
<i>pGrdSystemTime</i>	указатель на структуру TGrdSystemTime : <i>typedef struct { WORD wYear; // The year (2000 - 2099) WORD wMonth; // The month (January = 1, February = 2, ...) WORD wDayOfWeek; // The day of the week (Sunday = 0, Monday = 1, ...) WORD wDay; // The day of the month (1-31) WORD wHour; // The hour (0-23) WORD wMinute; // The minute (0-59) WORD wSecond; // The second (0-59) WORD wMilliseconds; // The millisecond (0-999) } TGrdSystemTime;</i>
<i>pReserved</i>	зарезервировано, должно быть равно NULL
GrdE_OK	нет ошибок
GrdE_NeedInitialization	требуется инициализация API (вызов GrdStartup)
GrdE_InvalidHandle	недействительный хэндл
GrdE_NoService	для алгоритма/ячейки сервис не предусмотрен
GrdE_InvalidArg	недопустимый параметр при вызове функции

Функция **GrdGetTime** возвращает время микросхемы таймера ключа Guardant Time. Для работы требуется предварительно установить **Private Read Code** при помощи функции [GrdSetAccessCodes](#).

Изменение времени ключа (в случае, если оно не защищено при выполнении [GrdProtect](#)), влияет на работу механизма **DeadTime** (или соответственно активированного **LifeTime**). К примеру, если установить время жизни - 1 год и 1 минута, активировать алгоритм, переставить дату на год вперед, то останется ~1 минута, аналогично наоборот, т.к. в ключе хранится именно дата деактивации алгоритма.

C#

```
public static GrdE GrdGetTime(Handle grdHandle, out GrdSystemTime systemTime)
```

grdHandle [in]

Тип: [Handle](#)

Нэндл, через который будет выполнена данная операция.

systemTime [out]

Тип: [GrdSystemTime](#)

Указатель на структуру GrdSystemTime

GrdE.OK	нет ошибок
GrdE.NeedInitialization	требуется инициализация API (вызов GrdStartup)
GrdE.InvalidHandle	недействительный хэндл
GrdE.NoService	для алгоритма/ячейки сервис не предусмотрен
GrdE.InvalidArg	недопустимый параметр при вызове функции

Метод **GrdGetTime** возвращает время микросхемы таймера ключа Guardant Stealth III Time. Для работы требуется предварительно установить **Private Read Code** при помощи метода [GrdSetAccessCodes](#).

Изменение времени ключа (в случае, если оно не защищено при выполнении [GrdProtect](#)), влияет на работу механизма **DeadTime** (или соответственно активированного **LifeTime**). К примеру, если установить время жизни - 1 год и 1 минута, активировать алгоритм, переставить дату на год вперед, то останется ~1 минута, аналогично наоборот, т.к. в ключе хранится именно дата деактивации алгоритма.

Java

```
public static GrdE GrdGetTime(Handle grdHandle, GrdSystemTime systemTime)
```

grdHandle [in]

Тип: [Handle](#)

Хэндл, через который будет выполнена данная операция.

systemTime [in]

Тип: [GrdSystemTime](#)

Указатель на структуру GrdSystemTime

GrdE.OK	нет ошибок
GrdE.NeedInitialization	требуется инициализация API (вызов GrdStartup)
GrdE.InvalidHandle	недействительный хэндл
GrdE.NoService	для алгоритма/ячейки сервис не предусмотрен
GrdE.InvalidArg	недопустимый параметр при вызове функции

Метод **GrdGetTime** возвращает время микросхемы таймера ключа Guardant Stealth III Time. Для работы требуется предварительно установить **Private Read Code** при помощи метода [GrdSetAccessCodes](#).

Изменение времени ключа (в случае, если оно не защищено при выполнении [GrdProtect](#)), влияет на работу механизма **DeadTime** (или соответственно активированного **LifeTime**). К примеру, если установить время жизни - 1 год и 1 минута, активировать алгоритм, переставить дату на год вперед, то останется ~1 минута, аналогично наоборот, т.к. в ключе хранится именно дата деактивации алгоритма.