Slave Control API > BS2_GetOsdpStandardDeviceCapability

BS2_GetOsdpStandardDeviceCapability

[+ 2.8.4] CoreStation40 OSDP   .

```
#include "BS_API.h"

int BS2_GetOsdpStandardDeviceCapability(void* context, uint32_t
osdpDeviceId, BS2OsdpStandardDeviceCapability* capability);
```

> BS2OsdpStandardDeviceCapability

- [In] *context* : Context
- [In] *osdpDeviceId* : OSDP
- [Out] *capability* :

BS_SDK_SUCCESS   ,

.

BS2_AddOsdpStandardDevice
BS2_GetOsdpStandardDevice
BS2_GetAvailableOsdpStandardDevice
BS2_UpdateOsdpStandardDevice
BS2_RemoveOsdpStandardDevice
BS2_GetOsdpStandardDeviceCapability
BS2_SetOsdpStandardDeviceSecurityKey

(C++)

sample_bs2_getosdpstandarddevicecapability.cpp

```
BS2OsdpStandardDeviceCapability capability = { , };

BS2_DEVICE_ID id = Utility::selectSlaveID();
int sdkResult = BS2_GetOsdpStandardDeviceCapability(context_, id,
&capability);
if (BS_SDK_SUCCESS != sdkResult)
    printf("BS2_GetOsdpStandardDeviceCapability call failed: %d",
sdkResult);

return sdkResult;
```

(C#)

sample_bs2_getosdpstandarddevicecapability.cs

```
UInt32 id = Util.GetInputSlaveID();

BS2OsdpStandardDeviceCapability capability;
BS2ErrorCode result =
(BS2ErrorCode)API.BS2_GetOsdpStandardDeviceCapability(sdkContext, id,
out capability);
if (result != BS2ErrorCode.BS_SDK_SUCCESS)
    Console.WriteLine("Got error({0}).", result);
else
    Console.WriteLine("Call success.");

return result;
```