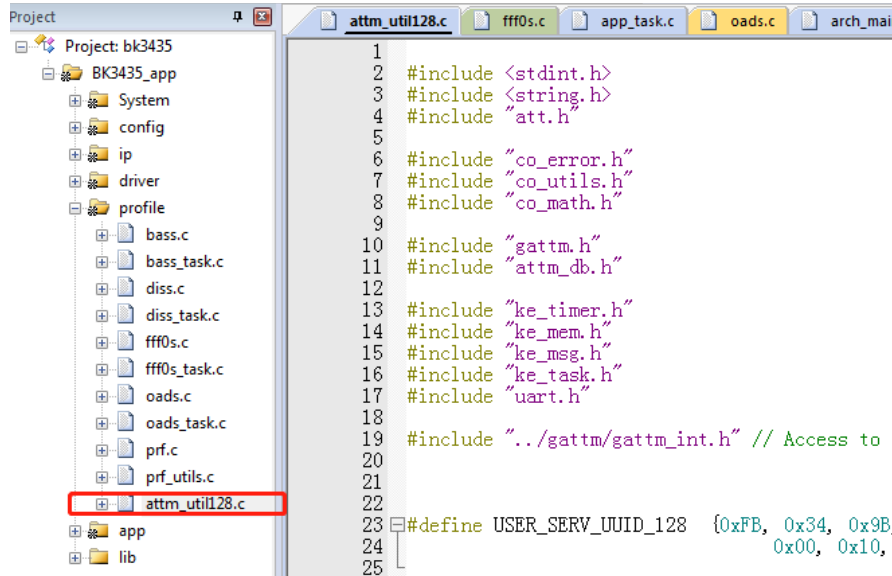


说明：该文档以修改fff0服务为例。

1. 在SDK工程里添加attm\_util128.c文件。



2. 在fff0s.c文件下的头文件声明#include "attm\_util128.h"。

3. 在fff0s.c文件下的fff0\_att\_db列表里做如下修改。

```
/// Full FFF0 Database Description - Used to add attributes into the database
const struct attm_desc fff0_att_db[FFF0_IDX_NB] =
{
    // FFF0 Service Declaration
    [FFF0_IDX_SVC] = {ATT_DECL_PRIMARY_SERVICE, PERM(RD, ENABLE), 0, 0},
    [FFF0_IDX_FFF2_LVL_CHAR] = {ATT_DECL_CHARACTERISTIC, PERM(RD, ENABLE), 0, 0},
    // Characteristic Value
    [FFF0_IDX_FFF2_LVL_VAL] = {ATT_USER_SERVER_CHAR_FFF2, PERM(WRITE_COMMAND, ENABLE), PERM(UUID_LEN, UUID_128) | PERM(RI, ENABLE) | 0x10, FFF0_FFF2_DATA_LEN * sizeof(uint8_t)},
    // fff0 Level Characteristic Declaration
    [FFF0_IDX_FFF1_LVL_CHAR] = {ATT_DECL_CHARACTERISTIC, PERM(RD, ENABLE), 0, 0},
    // fff0 Level Characteristic Value
    [FFF0_IDX_FFF1_LVL_VAL] = {ATT_USER_SERVER_CHAR_FFF1, PERM(WRITE_COMMAND, ENABLE), PERM(UUID_LEN, UUID_128) | PERM(RI, ENABLE) | 0x10, FFF0_FFF1_DATA_LEN * sizeof(uint8_t)},
    // fff0 Level Characteristic - Client Characteristic Configuration Descriptor
    [FFF0_IDX_FFF1_LVL_NTF_CFG] = {ATT_DESC_CLIENT_CHAR_CFG, PERM(RD, ENABLE) | PERM(WRITE_REQ, ENABLE), 0, 0},
};
/// Macro used to retrieve permission value from access and rights on attribute.
```

4. 在fff0s.c文件下的fff0s\_init函数做如下修改。

```
static uint8_t fff0s_init (struct prf_task_env* env, uint16_t* start_hdl, uint16_t app_task, uint8_t
{
    uint16_t shdl;
    struct fff0s_env_tag* fff0s_env = NULL;
    // Status
    uint8_t status = GAP_ERR_NO_ERROR;
    //----- allocate memory required for the profile -----
    fff0s_env = (struct fff0s_env_tag*) ke_malloc(sizeof(struct fff0s_env_tag), KE_MEM_ATT_DB);
    memset(fff0s_env, 0, sizeof(struct fff0s_env_tag));
    // Service content flag
    uint8_t cfg_flag = FFF0S_CFG_FLAG_MANDATORY_MASK;
    // Save database configuration
    fff0s_env->features |= (params->features);
    // Check if notifications are supported
    if (params->features == FFF0_FFF1_LVL_NTF_SUP)
    {
        cfg_flag |= FFF0S_CFG_FLAG_NTF_SUP_MASK;
    }
    shdl = *start_hdl;
    //Create FFF0 in the DB
    //----- create the attribute database for the profile -----
    status = attm_util_svc_create_db128(&(shdl), ATT_USER_SERVER_FFF0, (uint8_t*)&cfg_flag,
    FFF0S_IDX_NB, NULL, env->task, &fff0_att_db[0],
    (sec_lvl & (PERM_MASK_SVC_DIS | PERM_MASK_SVC_AUTH | PERM_MASK_SVC_EKS)));
    //Set optional permissions
    if (status == GAP_ERR_NO_ERROR)
    {
        //Set optional permissions
        if (params->features == FFF0_FFF1_LVL_NTF_SUP)
        {
            // Battery Level characteristic value permissions
            uint16_t perm = PERM(RD, ENABLE) | PERM(NTF, ENABLE);
            attm_att_set_permission(shdl + FFF0S_IDX_FFF1_LVL_VAL, perm, 0);
        }
    }
    //----- Update profile task information -----
```

5. 修改后的UUID如下：

### Unknown Service

UUID: 0000fff0-0000-1000-8000-00805f9b34fb

PRIMARY SERVICE

### Unknown Characteristic

UUID: fff284cf-f7e3-55b4-6c4c-9fd140100a16

Properties: WRITE NO RESPONSE



### Unknown Characteristic

UUID: fff184cf-f7e3-55b4-6c4c-9fd140100a16

Properties: NOTIFY, READ



### Descriptors:

Client Characteristic Configuration

UUID: 0x2902

