

|                   |   |
|-------------------|---|
| <b>Face API</b>   | 1 |
| <b>FaceEx API</b> | 1 |
| .....             | 1 |
| BS2Face           | 1 |
| BS2AuthGroup      | 2 |
| BS2TemplateEx     | 2 |
| BS2FaceEx         | 3 |

# Face API

API .

- [BS2\\_ScanFace](#): FaceStation2 .
- [BS2\\_GetAuthGroup](#): 가 .
- [BS2\\_GetAllAuthGroup](#): 가 . \*
- [BS2\\_SetAuthGroup](#): .
- [BS2\\_RemoveAuthGroup](#): .
- [BS2\\_RemoveAllAuthGroup](#): .

## FaceEx API

- [BS2\\_ScanFaceEx](#): FaceStation F2 .  
[ + 2.7.1]
- [BS2\\_ExtractTemplateFaceEx](#): FaceStation F2 가 template .  
[ + 2.7.1]

### BS2Face

```
typedef struct {
    uint8_t faceIndex;
    uint8_t numOfTemplate;
    uint8_t flag;
    uint8_t reserved;

    uint16_t imageLen;
    uint8_t reserved2[2];

    uint8_t imageData[BS2_FACE_IMAGE_SIZE];
    uint8_t templateData[BS2_TEMPLATE_PER_FACE][BS2_FACE_TEMPLATE_LENGTH];
} BS2Face;
```

1. *faceIndex*

2. *numOfTemplate*

3. *flag*

4. *reserved*

5. *imageLen*

6. *reserved2*

4. *imageData*

5. *data*

## BS2AuthGroup

```
typedef struct {  
    BS2_AUTH_GROUP_ID    id;  
    char                 name[BS2_MAX_AUTH_GROUP_NAME_LEN];  
    uint8_t              reserved[32];  
} BS2AuthGroup;
```

1. *id*

2. *name*  
BioStar

3. *reserved*

## BS2TemplateEx

```
typedef struct {  
    uint8_t      data[552];  
    uint8_t      isIR;  
    uint8_t      reserved[3];  
} BS2TemplateEx;
```

FaceStation F2

1. *data*  
IR      visual

2. *isIR*  
IR              true, visual              false

3. *reserved*

## BS2FaceEx

```
typedef struct {
    uint8_t      faceIndex;
    uint8_t      numOfTemplate;
    uint8_t      flag;
    uint8_t      reserved;

    uint32_t      imageLen;
    union {
        struct {
            uint16_t irImageLen;
            uint8_t  unused[6];          ///< 6 bytes (packing)
            uint8_t  imageData[BS2_MAX_WARPED_IMAGE_LENGTH];    ///<
40 * 1024 bytes
            uint8_t  irImageData[BS2_MAX_WARPED_IR_IMAGE_LENGTH];    ///<
30 * 1024 bytes
            BS2TemplateEx templateEx[BS2_MAX_TEMPLATES_PER_FACE_EX];    ///<
20 * 556 bytes
        };

        uint8_t      *rawImageData;
    };
} BS2FaceEx;
```

## FaceStation F2

### 1. *faceIndex*

## 2. *numOfTemplate*

Visual, IR

### 3. *flag*

가 WARP

WARP ,

flag가 1

flag가 0

가 WARP

```
flag    BS2 FACE EX FLAG NONE(0)
```

imageLen

rawImageData 가

## WARP

struct

rawImageData struct or union

|                         |      |
|-------------------------|------|
|                         |      |
| BS2_FACE_EX_FLAG_NONE   | 0x00 |
| BS2_FACE_EX_FLAG_WARPED | 0x01 |
| BS2_FACE_EX_FLAG_ALL    | 0xFF |

4. *reserved*

5. *imageLen*

6. *irlImageLen*

IR

7. *unused*

. (packing )

8. *imageData*

WARP

. *rawImageData*

9. *irlImageData*

IR

. WARP

IR

10. *templateEx*

Visual IR

11. *rawImageData*

WARP

. (JPG

,

250 \* 250)

From:

<http://kb.supremainc.com/bs2sdk/> - **BioStar 2 Device SDK**

Permanent link:

[http://kb.supremainc.com/bs2sdk/doku.php?id=ko:face\\_api&rev=1612769292](http://kb.supremainc.com/bs2sdk/doku.php?id=ko:face_api&rev=1612769292)

Last update: **2021/02/08 16:28**