목차

Lift Control API 1
- BS2Lift 1
- BS2LiftFloor 3
- BS2FloorStatus 3
- BS2LiftSensor 4
- BS2LiftAlarm 4
- BS2LiftStatus 5
- BS2FloorLevel 5
- BS2FloorSchedule 6

Lift Control API

Lift를 관리하기 위한 API로 floorLevel 관리를 통해 Lift 제한 및 관리할 수 있습니다.

- BS2_GetLift: 주어진 Lift를 가져옵니다.
- BS2_GetAllLift: 모든 Lift를 가져옵니다.
- BS2_GetLiftStatus: 주어진 Lift의 상태 정보를 가져옵니다
- BS2_GetAllLiftStatus: 모든 Lift의 상태 정보를 가져옵니다
- BS2_SetLift: Lift을 설정합니다.
- BS2_SetLiftAlarm: Lift의 경보 상태를 갱신합니다.
- BS2_RemoveLift: 주어진 Lift를 제거합니다.
- BS2_RemoveAllLift: 모든 Lift를 제거합니다.
- BS2_ReleaseFloor: Floor의 우선 순위를 제거합니다.
- BS2_ActivateFloor: Floor가 허용될 때의 우선 순위를 설정하며, 허용되지 않을 때의 우선 순위보다 높아야 Floor에 접근 가능합니다.
- BS2_DeActivateFloor: Floor가 허용되지 않을 때의 우선 순위를 설정하며, 허용될 때의 우선 순위보다 높아야 Floor에 접근을 막을 수 있습니다.
- BS2_GetFloorLevel: 주어진 Floor 레벨을 가져옵니다.
- BS2_GetAllFloorLevel: 모든 Floor 레벨을 가져옵니다.
- BS2_SetFloorLevel: Floor 레벨을 설정합니다.
- BS2_RemoveFloorLevel: 주어진 Floor 레벨을 제거합니다.
- BS2_RemoveAllFloorLevel: 모든 Floor 레벨을 제거합니다.

BS2Lift

```
typedef struct {
    BS2_LIFT_ID         liftID;
    char             name[BS2_MAX_LIFT_NAME_LEN];

    BS2_DEVICE_ID            deviceID[BS2_MAX_DEVICES_ON_LIFT];

    uint32_t         activateTimeout;
    uint32_t         dualAuthTimeout;

    uint8_t                    numFloors;
    uint8_t                    numDualAuthApprovalGroups;
    BS2_DUAL_AUTH_APPROVAL    dualAuthApprovalType;
    BS2_BOOL             tamperOn;

    BS2_BOOL                dualAuthRequired[BS2_MAX_DEVICES_ON_LIFT];
    BS2_SCHEDULE_ID        dualAuthScheduleID;

    BS2LiftFloor         floor[BS2_MAX_FLOORS_ON_LIFT];
    BS2_ACCESS_GROUP_ID
dualAuthApprovalGroupID[BS2_MAX_DUAL_AUTH_APPROVAL_GROUP_ON_LIFT];

    BS2LiftAlarm            alarm[BS2_MAX_ALARMS_ON_LIFT];
    BS2LiftAlarm            tamper;
```

```
    BS2_LIFT_ALARM_FLAG    alarmFlags;
    uint8_t                 reserved[3];
} BS2Lift;
```

1. *liftID*
lift 식별자입니다.

2. *name*
BioStar 애플리케이션에서 보여줄 Lift 이름입니다.

3. *deviceID*
lift 장치와 연결된 장치 식별자입니다.

4. *activateTimeout*
lift가 열린 후 자동으로 닫힐때까지 걸리는 시간으로 초 단위입니다.

5. *dualAuthTimeout*
1번째 사용자의 인증이 끝나고 2번째 사용자의 인증을 시도하기까지 대기하는 시간으로 초 단위입니다.

6. *numFloors*
lift에 속한 Floor의 개수입니다.

7. *numDualAuthApprovalGroups*
이중 출입 권한을 가지는 그룹의 총 개수입니다.

8. *dualAuthApprovalType*
출입을 허용할 때 사용자의 출입 그룹이 이중 인증 출입 권한을 가지는 그룹에 속한지 검사하는 flag입니다.

값	설명
0	없음
1	마지막 사용자의 권한 확인

9. *tamperOn*
현재 tamper 상태 여부입니다.

10. *dualAuthRequired*
장치의 이중 인증 필요 여부입니다.

11. *dualAuthScheduleID*
이중 인증이 필요한 타임 스케줄로 이중 인증이 필요 없다면 0을, 항상 필요하면 1을, 아니면 적절한 타임 스케줄 식별자를 지정할 수 있습니다.

12. *floor*
lift의 floor 정보입니다. 최대 255까지 설정할 수 있습니다.

13. *dualAuthApprovalGroupID*
이중 출입 권한을 가지는 그룹을 나열한 리스트로 최대 16까지 설정할 수 있습니다.

14. *alarm*
lift의 sensor가 감지되었을 때 동작할 경보로 최대2개까지 설정할 수 있습니다.

15. *tamper*

lift의 tamper가 감지되었을 때 동작할 경보 정보입니다.

16. *alarmFlags*
경보 flag입니다.

17. *reserved*
예약된 공간입니다.

BS2LiftFloor

```
typedef struct {
    BS2_DEVICE_ID    deviceID;
    uint8_t         port;
    BS2FloorStatus  status;
} BS2LiftFloor;
```

1. *deviceID*
장치 식별자입니다.

2. *port*
장치의 입력 포트입니다

3. *status*
Floor 상태 정보입니다.

BS2FloorStatus

```
typedef struct {
    BS2_BOOL         activated;
    BS2_FLOOR_FLAG        activateFlags;
    BS2_FLOOR_FLAG        deactivateFlags;
} BS2FloorStatus;
```

1. *activated*
대기상태에서 열림과 닫힘의 상태 정보입니다.

2. *activateFlags*
floor가 열릴 때의 우선 순위로 닫힐 때의 우선 순위보다 낮으면 수행되지 않습니다. 예를 들어, 관리자 권한으로 floor를 닫은 상태라면(OPERATOR), 모든 사용자의 출입이 허용되지 않습니다. 유의해야 할 사항으로 activateFlags와 deactivateFlags는 기본 우선 순위(NONE)를 제외하고는 동일 우선 순위를 가질 수 없습니다.

값	설명	우선순위
0	기본	보통
1	일정	약간 높음
4	관리자	높음
2	긴급	매우 높음

3. *deactivateFlags*
floor가 닫힐 때의 우선 순위로 열릴 때의 우선 순위보다 낮으면 수행되지 않습니다

값	설명	우선순위
0	기본	보통
1	일정	약간 높음
4	관리자	높음
2	긴급	매우 높음

BS2LiftSensor

```
typedef struct {
    BS2_DEVICE_ID            deviceID;
    uint8_t            port;
    BS2_SWITCH_TYPE        switchType;
    uint16_t        duration;
    BS2_SCHEDULE_ID        scheduleID;
} BS2LiftSensor;
```

1. *deviceID*
장치 식별자입니다.

2. *port*
장치의 입력 포트입니다.

3. *switchType*
입력 신호 유형입니다.

값	설명
0	대기 상태에서 열림 상태로 유지
1	대기 상태에서 닫힘 상태로 유지

4. *duration*
입력 신호를 화재 경보 상태로 판단하기까지 걸리는 시간으로 millisecond 단위입니다.

5. *scheduleID*
lift를 동작하기 위한 타임 스케줄입니다.

BS2LiftAlarm

```
typedef struct {
    BS2LiftSensor sensor;
    BS2Action action;
} BS2LiftAlarm;
```

1. *sensor*
lift의 열림/닫힘 상태를 감지하는 센서입니다

2. *action*
lift가 수행하는 action입니다.

BS2LiftStatus

```
typedef struct {
    BS2_LIFT_ID          liftID;
    uint16_t             numFloors;
    BS2_LIFT_ALARM_FLAG  alarmFlags;
    BS2_BOOL             tamperOn;
    BS2FloorStatus       floors[BS2_MAX_FLOORS_ON_LIFT];
} BS2LiftStatus;
```

1. *liftID*
lift 식별자입니다.

2. *numFloors*
lift에 속한 Floor 개수입니다.

3. *alarmFlags*
현재 lift의 경보 상태 정보입니다.

값	설명
0	경보 없음
1	첫번째 경보
2	두번째 경보
4	Tamper

4. *tamperOn*
현재 tamper 상태 정보입니다.

5. *floors*
floor를 나열한 리스트로 최대 255까지 설정할 수 있습니다.

BS2FloorLevel

```
typedef struct {
    BS2_FLOOR_LEVEL_ID        id;        // id >= 32768 (BS2_ACCESS_LEVEL_ID < 32768)
    char                  name[BS2_MAX_FLOOR_LEVEL_NAME_LEN];
    uint8_t                  numFloorSchedules;
    uint8_t                  reserved[3];
    BS2FloorSchedule          floorSchedules[BS2_MAX_FLOOR_LEVEL_ITEMS];
} BS2FloorLevel;
```

1. *id*

Floor 레벨 식별자입니다. id >= 32768 (BS2_ACCESS_LEVEL_ID < 32768)

2. *name*
BioStar 애플리케이션에서 보여줄 Floor 레벨 이름입니다.

3. *numFloorSchedules*
Floor 레벨에 속한 Floor 타임 스케줄 개수입니다.

4. *reserved*
예약된 공간입니다.

5. *floorSchedules*
Floor 레벨에 속한 Floor 타임 스케줄 리스트입니다

BS2FloorSchedule

```
typedef struct {
    BS2_LIFT_ID        liftID;
    uint16_t        floorIndex;
    uint8_t            reserved[2];
    BS2_SCHEDULE_ID            scheduleID;
} BS2FloorSchedule;
```

1. *liftID*
lift 식별자입니다.

2. *floorIndex*
floor 식별자입니다.

3. *reserved*
예약된 공간입니다.

4. *scheduleID*
타임 스케줄 식별자입니다.

From:
https://kb.supremainc.com/kbtest/ - **BioStar Device SDK**

Permanent link:
https://kb.supremainc.com/kbtest/doku.php?id=ko:lift_control_api&rev=1488868224

Last update: **2017/03/07 15:30**