

Zone Control API 1

..... 1

..... 1

..... 2

/ 2

..... 2

..... 3

Ethernet 3

..... 4

OnCheckGlobalAPBViolation 4

..... 4

BS2ZoneStatus 4

BS2ApbMember 5

BS2TimedApbMember 5

BS2FireSensor 5

BS2AntiPassbackZone 6

BS2TimedAntiPassbackZone 7

BS2FireAlarmZone 8

BS2ScheduledLockUnlockZone 9

BS2IntrusionAlarmZone 11

BS2IntrusionAlarmZoneBlob 12

BS2InterlockZone 13

BS2InterlockZoneBlob 13

BS2DeviceZoneEntranceLimitMaster 14

BS2DeviceZoneEntranceLimitMember 15

BS2DeviceZoneFireAlarmMaster 16

BS2DeviceZoneFireAlarmMember 16

BS2DeviceZoneFireAlarmMemberInfo 17

BS2DeviceZoneFireSensor 18

BS2DeviceZone 18

BS2DeviceZoneAGEntranceLimit 19

Zone Control API

, BioStart

4

가

가

(,)

- [BS2_GetAntiPassbackZone:](#)
- [BS2_GetAllAntiPassbackZone:](#)
- [BS2_GetAntiPassbackZoneStatus:](#)
- [BS2_GetAllAntiPassbackZoneStatus:](#)
- [BS2_SetAntiPassbackZone:](#)
- [BS2_SetAntiPassbackZoneAlarm:](#)
- [BS2_RemoveAntiPassbackZone:](#)
- [BS2_RemoveAllAntiPassbackZone:](#)
- [BS2_ClearAntiPassbackZoneStatus:](#)
- [BS2_ClearAllAntiPassbackZoneStatus:](#)
- [BS2_SetCheckGlobalAPBViolationHandler:](#)
- [BS2_CheckGlobalAPBViolation:](#)

가

가

가

가

가

가

(,)

- [BS2_GetTimedAntiPassbackZone:](#)
- [BS2_GetAllTimedAntiPassbackZone:](#)
- [BS2_GetTimedAntiPassbackZoneStatus:](#)
- [BS2_GetAllTimedAntiPassbackZoneStatus:](#)
- [BS2_SetTimedAntiPassbackZone:](#)
- [BS2_SetTimedAntiPassbackZoneAlarm:](#)
- [BS2_RemoveTimedAntiPassbackZone:](#)
- [BS2_RemoveAllTimedAntiPassbackZone:](#)
- [BS2_ClearTimedAntiPassbackZoneStatus:](#)
- [BS2_ClearAllTimedAntiPassbackZoneStatus:](#)

가

가

가

가

가 BioStar

BioStar

- [BS2_GetFireAlarmZone:](#)
- [BS2_GetAllFireAlarmZone:](#)
- [BS2_GetFireAlarmZoneStatus:](#)
- [BS2_GetAllFireAlarmZoneStatus:](#)
- [BS2_SetFireAlarmZone:](#)
- [BS2_SetFireAlarmZoneAlarm:](#)
- [BS2_RemoveFireAlarmZone:](#)
- [BS2_RemoveAllFireAlarmZone:](#)

가
가

가
가

/

/

- [BS2_GetScheduledLockUnlockZone:](#)
- [BS2_GetAllScheduledLockUnlockZone:](#)
- [BS2_GetScheduledLockUnlockZoneStatus:](#)
- [BS2_GetAllScheduledLockUnlockZoneStatus:](#)
- [BS2_SetScheduledLockUnlockZone:](#)
- [BS2_SetScheduledLockUnlockZoneAlarm:](#)
- [BS2_RemoveScheduledLockUnlockZone:](#)
- [BS2_RemoveAllScheduledLockUnlockZone:](#)

/
/

가
가

가
가

가 BioStar

가

BioStar

- [BS2_GetIntrusionAlarmZone:](#)
- [BS2_GetIntrusionAlarmZoneStatus:](#)
- [BS2_GetAllIntrusionAlarmZoneStatus:](#)
- [BS2_SetIntrusionAlarmZone:](#)
- [BS2_SetIntrusionAlarmZoneAlarm:](#)
- [BS2_RemoveIntrusionAlarmZone:](#)
- [BS2_RemoveAllIntrusionAlarmZone:](#)
- [BS2_SetIntrusionAlarmZoneArm:](#)

가

가
가

/

[CoreStation]

CoreStation

- [BS2_GetInterlockZone:](#) 가
- [BS2_GetInterlockZoneStatus:](#) 가
- [BS2_GetAllInterlockZoneStatus:](#) 가
- [BS2_SetInterlockZone:](#)
- [BS2_SetInterlockZoneAlarm:](#)
- [BS2_RemoveInterlockZone:](#)
- [BS2_RemoveAllInterlockZone:](#)

Ethernet

Zone Master BioStar V2.x 가 Zone 가 (Master ↔ Member)
 Ethernet TCP 1.x Entrance Limit, Fire Alarm Zone

- [BS2_GetDeviceZone:](#) Ethernet 가
- [BS2_GetAllDeviceZone:](#) Ethernet 가
- [BS2_SetDeviceZone:](#) Ethernet
- [BS2_RemoveDeviceZone:](#) Ethernet
- [BS2_RemoveAllDeviceZone:](#) Ethernet
- [BS2_SetDeviceZoneAlarm:](#) Ethernet
- [BS2_ClearDeviceZoneAccessRecord:](#) Ethernet
- [BS2_ClearAllDeviceZoneAccessRecord:](#) Ethernet
- [BS2_GetAccessGroupEntranceLimit:](#) 가
- [BS2_GetAllAccessGroupEntranceLimit:](#) 가
- [BS2_SetAccessGroupEntranceLimit:](#)
- [BS2_RemoveAccessGroupEntranceLimit:](#)
- [BS2_RemoveAllAccessGroupEntranceLimit:](#)
- [BS2_GetDeviceZoneAGEntranceLimit:](#) Ethernet Access Group 가
- [BS2_GetAllDeviceZoneAGEntranceLimit:](#) Ethernet Access Group 가
- [BS2_SetDeviceZoneAGEntranceLimit:](#) Ethernet Access Group
- [BS2_RemoveDeviceZoneAGEntranceLimit:](#) Ethernet Access Group
- [BS2_RemoveAllDeviceZoneAGEntranceLimit:](#) Ethernet Access Group

OnCheckGlobalAPBViolation

```
typedef void (*OnCheckGlobalAPBViolation)(uint32_t deviceId, uint16_t seq, const char* userID_1, const char* userID_2, bool isDualAuth);
```

- 1. *OnCheckGlobalAPBViolation* 가

BS2ZoneStatus

```
typedef struct {  
    uint32_t id;  
    uint8_t status;  
    uint8_t disabled;  
    uint8_t reserved[6];  
} BS2ZoneStatus;
```

- 1. *id*
- 2. *status*

0	
1	
2	scheduled lock
4	scheduled unlock

- 3. *disabled* flag
- 4. *reserved*

BS2ApbMember

```
typedef struct {
    uint32_t deviceID;
    uint8_t type;
    uint8_t reserved[3];
} BS2ApbMember;
```

1. *deviceID*

2. *type*

APB reader

-1	
0	
1	

3. *reserved*

BS2TimedApbMember

```
typedef struct {
    uint32_t deviceID;
    uint8_t reserved[4];
} BS2TimedApbMember;
```

1. *deviceID*

2. *reserved*

BS2FireSensor

```
typedef struct {
    uint32_t deviceID;
    uint8_t port;
    uint8_t switchType;
    uint8_t duration;
} BS2FireSensor ;
```

1. *deviceID*

2. *port*

3. switchType

0	
1	

4. duration

millisecond

BS2AntiPassbackZone

```
typedef struct {
    uint32_t zoneID;
    char name[BS2_MAX_ZONE_NAME_LEN];
    uint8_t type;
    uint8_t numReaders;
    uint8_t numBypassGroups;
    uint8_t disabled;
    uint8_t alarmed;
    uint8_t reserved[3];
    uint32_t resetDuration;
    BS2Action alarm[BS2_MAX_APB_ALARM_ACTION];
    BS2ApbMember readers[BS2_MAX_READERS_PER_APB_ZONE];
    uint8_t reserved2[512];
    uint32_t bypassGroupIDs[BS2_MAX_BYPASS_GROUPS_PER_APB_ZONE];
} BS2AntiPassbackZone;
```

1. zoneID

1 가

	가

2. name

BioStar

3. type

0	Hard APB(가)
1	Soft APB()

4. *numReaders*

APB reader

5. *numBypassGroups*

APB

6. *disabled*

flag

7. *alarmed*8. *reserved*9. *resetDuration*

가 APB

0

BioStar

10. *alarm*

가 APB

5

11. *readers*

64

12. *reserved2*13. *bypassGroupIDs*

APB

16

BS2TimedAntiPassbackZone

```

typedef struct {
    uint32_t zoneID;
    char name[BS2_MAX_ZONE_NAME_LEN];
    uint8_t type;
    uint8_t numReaders;
    uint8_t numBypassGroups;
    uint8_t disabled;
    uint8_t alarmed;
    uint8_t reserved[3];
    uint32_t resetDuration;
    BS2Action alarm[BS2_MAX_TIMED_APB_ALARM_ACTION];
    BS2TimedApbMember readers[BS2_MAX_READERS_PER_TIMED_APB_ZONE];
    uint8_t reserved2[320];
    uint32_t bypassGroupIDs[BS2_MAX_BYPASS_GROUPS_PER_TIMED_APB_ZONE];
} BS2TimedAntiPassbackZone;

```

1. *zoneID*

1 가

2. name

BioStar

3. type

0	Hard APB(가)
1	Soft APB()

4. numReaders

reader

5. numBypassGroups

6. disabled

flag

7. alarmed

8. reserved

9. resetDuration

가

0

BioStar

10. alarm

가

5

11. readers

64

12. reserved2

13. bypassGroupIDs

16

BS2FireAlarmZone

```
typedef struct {
    uint32_t zoneID;
    char name[BS2_MAX_ZONE_NAME_LEN];
};
```

```

uint8_t numSensors;
uint8_t numDoors;
uint8_t disabled;
uint8_t alarmed;
uint8_t reserved[8];
BS2FireSensor sensor[BS2_MAX_FIRE_SENSORS_PER_FIRE_ALARM_ZONE];
BS2Action alarm[BS2_MAX_FIRE_ALARM_ACTION];
uint8_t reserved2[32];
uint32_t doorIDs[BS2_MAX_DOORS_PER_FIRE_ALARM_ZONE];
} BS2FireAlarmZone;

```

1. zoneID

1 가

2. name

BioStar

3. numSensors

4. numDoors

5. alarmed

6. disabled

flag

7. reserved

8. sensor

8

9. alarm

5

10. reserved2

11. doorIDs

32

BS2ScheduledLockUnlockZone

```

typedef struct {
uint32_t zoneID;
char name[BS2_MAX_ZONE_NAME_LEN];
uint32_t lockScheduleID;
uint32_t unlockScheduleID;
uint8_t numDoors;
}

```

```

uint8_t numBypassGroups;
uint8_t numUnlockGroups;
uint8_t bidirectionalLock;
uint8_t disabled;
uint8_t alarmed;
uint8_t reserved[6];
BS2Action alarm[BS2_MAX_SCHEDULED_LOCK_UNLOCK_ALARM_ACTION];
uint8_t reserved2[32];
uint32_t doorIDs[BS2_MAX_DOORS_IN_SCHEDULED_LOCK_UNLOCK_ZONE];
uint32_t
bypassGroupIDs[BS2_MAX_BYPASS_GROUPS_IN_SCHEDULED_LOCK_UNLOCK_ZONE];
uint32_t
unlockGroupIDs[BS2_MAX_UNLOCK_GROUPS_IN_SCHEDULED_LOCK_UNLOCK_ZONE];
} BS2ScheduledLockUnlockZone;

```

1. zoneID

1 가

2. name

BioStar

3. lockScheduleID

4. unlockScheduleID

5. numDoors

6. numBypassGroups

7. numUnlockGroups

가

8. bidirectionalLock

9. disabled

flag

10. alarmed

11. reserved

12. alarm

5

13. reserved2

14. *doorIDs*

32

15. *bypassGroupIDs*

16

16. *unlockGroupIDs*

가

16

BS2IntrusionAlarmZone

```
typedef struct {
    uint32_t zoneID;
    char name[BS2_MAX_ZONE_NAME_LEN];
    uint8_t armDelay;
    uint8_t alarmDelay;
    uint8_t disabled;
    uint8_t reserved[1];
    uint8_t numReaders;
    uint8_t numInputs;
    uint8_t numOutputs;
    uint8_t numCards;
    uint8_t numDoors;
    uint8_t numGroups;
    uint8_t reserved2[10];
} BS2IntrusionAlarmZone;
```

1. *zoneID*

1

가

2. *name*

BioStar

3. *armDelay*

4. *alarmDelay*

5. *disabled*

flag

6. *reserved[1]*

7. *numReaders*

8. *numInputs*

9. *numOutputs*
10. *numCards*
11. *numDoors*
12. *numGroups*
13. *reserved*

BS2IntrusionAlarmZoneBlob

```
typedef struct {  
    BS2IntrusionAlarmZone IntrusionAlarmZone;  
    BS2AlarmZoneMember* memberObjs;  
    BS2AlarmZoneInput* inputObjs;  
    BS2AlarmZoneOutput* outputObjs;  
    BS2CSNCard* cardObjs;  
    BS2_D00R_ID* doorIDs;  
    BS2_ACCESS_GROUP_ID* groupIDs;  
} BS2IntrusionAlarmZoneBlob;
```

1. *IntrusionAlarmZone*

2. *memberObjs*

IntrusionAlarmZone.numReaders

3. *inputObjs*

IntrusionAlarmZone.numInputs

4. *outputObjs*

IntrusionAlarmZone.numOutputs

5. *cardObjs*

IntrusionAlarmZone.numCards

[Smartcard API](#)

6. *doorIDs*

IntrusionAlarmZone.numDoors

7. groupIDs

IntrusionAlarmZone.numGroups

BS2InterlockZone

```
typedef struct {
    uint32_t zoneID;
    char name[BS2_MAX_ZONE_NAME_LEN];
    uint8_t disabled;
    uint8_t numInputs;
    uint8_t numOutputs;
    uint8_t numDoors;
    uint8_t reserved[8];
} BS2InterlockZone;
```

1. zoneID

1 가

2. name

BioStar

3. disabled

flag

4. numInputs

5. numOutputs

6. numDoors

7. reserved

BS2InterlockZoneBlob

```
typedef struct {
    BS2InterlockZone InterlockZone;
    BS2InterlockZoneInput* inputObjs;
    BS2InterlockZoneOutput* outputObjs;
    BS2_D00R_ID* doorIDs;
} BS2InterlockZoneBlob;
```

1. InterlockZone

4. *entryLimitInterval_s*

5. *numEntranceLimit*

6. *numReaders*
reader

7. *numAlarm*

8. *numBypassGroups*

9. *maxEntry*

10. *periodStart_s*
가 ()

11. *periodEnd_s*
가 ()

12. *readers* 64

13. *alarm* 5

14. *bypassGroupIDs* 16

15. *reserved3*

BS2DeviceZoneEntranceLimitMember

```

typedef struct {
    uint16_t masterPort;
    BS2_DEVICE_ZONE_ENTRANCE_LIMIT_DISCONNECTED_ACTION_TYPE
actionInDisconnect;
    uint8_t reserved1[1];
    BS2_IPV4_ADDR masterIP;
} BS2DeviceZoneEntranceLimitMember;

```

1. *masterPort*
master port

2. *actionInDisconnect*
Disconnect

1	Soft EntranceLimit Disconnected action() 가
2	Hard EntranceLimit Disconnected action()

3. reserved1[3]

4. masterIP

master IP

BS2DeviceZoneFireAlarmMaster

```
typedef struct {
    char name[BS2_MAX_ZONE_NAME_LEN];
    uint8_t numReaders;
    uint8_t numAlarm;
    uint8_t reserved1[2];
    BS2DeviceZoneFireAlarmMemberInfo
    readers[BS2_MAX_READERS_PER_DEVICE_ZONE_FIRE_ALARM];
    BS2Action alarm[BS2_MAX_DEVICE_ZONE_FIRE_ALARM_ALARM_ACTION];
    uint8_t reserved2[8 * 40];
} BS2DeviceZoneFireAlarmMaster;
```

1. name

BioStar

2. numReaders

3. reserved1

4. readers

5. alarm

5

6. reserved2

BS2DeviceZoneFireAlarmMember

```
typedef struct {
    BS2_PORT masterPort;
    uint8_t reserved1[2];
}
```

```

    BS2_IPV4_ADDR masterIP;
    uint8_t numSensors;
    uint8_t numDoors;
    uint8_t reserved2[2];
    BS2DeviceZoneFireSensor
sensor[BS2_MAX_FIRE_SENSORS_PER_DEVICE_ZONE_FIRE_ALARM_MEMBER];
    union {
        BS2_DOOR_ID
doorIDs[BS2_MAX_DOORS_PER_DEVICE_ZONE_FIRE_ALARM_MEMBER];
        BS2_LIFT_ID
liftIDs[BS2_MAX_DOORS_PER_DEVICE_ZONE_FIRE_ALARM_MEMBER];
    };
} BS2DeviceZoneFireAlarmMember;

```

1. *masterPort*

master port .

2. *reserved1*

.

3. *masterIP*

master IP .

4. *numSensors*

.

5. *numDoors*

.

6. *reserved2*

.

7. *sensor*

8

.

8. *doorIDs*

8

.

9. *liftIDs*

Lift

8

.

BS2DeviceZoneFireAlarmMemberInfo

```

typedef struct {
    uint32_t readerID;
} BS2DeviceZoneFireAlarmMemberInfo;

```

1. *readerID*

.

BS2DeviceZoneFireSensor

```
typedef struct {
    uint32_t deviceID;
    uint8_t port;
    uint8_t switchType;
    uint16_t duration;
} BS2DeviceZoneFireSensor;
```

1. *deviceID*

2. *port*

3. *switchType*

0	
1	

4. *duration*

millisecond

BS2DeviceZone

```
typedef struct {
    uint32_t zoneID;
    uint8_t zoneType;
    uint8_t nodeType;
    uint8_t enable;
    uint8_t reserved[1];
    union {
        BS2DeviceZoneEntranceLimitMaster entranceLimitMaster;
        BS2DeviceZoneEntranceLimitMember entranceLimitMember;
        BS2DeviceZoneFireAlarmMaster fireAlarmMaster;
        BS2DeviceZoneFireAlarmMember fireAlarmMember;
    };
} BS2DeviceZone;
```

1. *zoneID*

1 가

2. *zoneType*

3. *nodeType*

- 4. *enable*
flag
- 5. *reserved[1]*

BS2DeviceZoneAGEntranceLimit

```
typedef struct {
    uint32_t zoneID;
    uint16_t numAGEntranceLimit;
    uint16_t reserved1;
    uint32_t periodStart_s[BS2_MAX_ENTRANCE_LIMIT_PER_ZONE];
    uint32_t periodEnd_s[BS2_MAX_ENTRANCE_LIMIT_PER_ZONE];
    uint16_t numEntry[BS2_MAX_ENTRANCE_LIMIT_PER_ZONE];
    uint16_t
maxEntry[BS2_MAX_ENTRANCE_LIMIT_PER_ZONE][BS2_MAX_ACCESS_GROUP_ENTRANCE_LIMI
T_PER_ENTRANCE_LIMIT];
    uint32_t
accessGroupID[BS2_MAX_ENTRANCE_LIMIT_PER_ZONE][BS2_MAX_ACCESS_GROUP_ENTRANCE
_LIMIT_PER_ENTRANCE_LIMIT];
} BS2DeviceZoneAGEntranceLimit;
```

- 1. *zoneID*
1 가
- 2. *numAGEntranceLimit*
- 3. *reserved1*
- 4. *periodStart_s*
가
- 5. *periodEnd_s*
가
- 6. *numEntry*
- 7. *maxEntry*
- 8. *accessGroupID*
16

From:

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

Permanent link:

https://kb.supremainc.com/kbtest/doku.php?id=ko:zone_control_api&rev=1521521703

Last update: **2018/03/20 13:55**