

ParkInfo

Packing Information Panel
Release 1.0 Draft 1

■ 문서 이력

버전	일자	작성자	문서 변경 내용
1.0	2024.04	Gil	최초 작성

내용

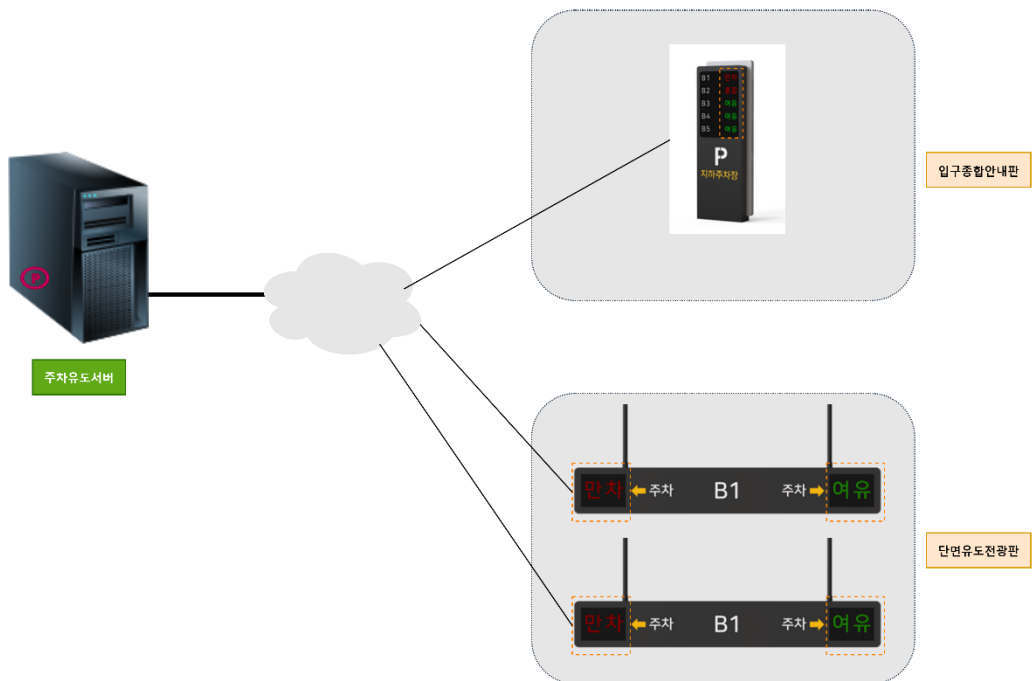
1. History	4
2. Concept	5
2.1 일반사항	5
2.2 용어정의 및 약어	5
2.3 입구 종합 안내판	6
2.4 단면 유도 전광판	7
3. parkinfo package	8
3.1 type EndPoint	8
3.1.1 func NewClient(addr string) *EndPoint	8
3.1.2 func (*EndPoint) IsConnected() bool	9
3.1.3 func (*EndPoint) Close	10
3.1.4 func (*EndPoint) Write (message Message) error	11
4. 예제	13
5. 사용자 참고 사항	14
5.1 Known Bug	14
5.2 개선 사항	14
6. 개발자 참고 사항	15

1. History

입구 종합 안내판, 단면 유도 전광판 제어를 위한 Package 정리한다.

입구 종합 안내판, 단면 유도 전광판은 전광판 프로토콜을 제공한다.

네트워크 구성은 아래와 같다.



2. Concept

2.1 일반사항

주차 통합 시스템에서 주차 안내판(입구 종합 안내판, 단면 유도 전광판)를 제어하는 패키지이다.
Package 명은 'parkinfo' 라고 한다.

2.2 용어정의 및 약어

입구 종합 안내판	주차장 진입위치에 각 층별 주차 가능 및 주차 현황을 표시
단면 유도 전광판	해당 층의 주차 가능 대수 및 주차 현황(여유, 만차, 혼잡)을 표시
주차 안내판	입구 종합 안내판, 단면 유도 전광판을 총칭

2.3 입구 종합 안내판

입구 종합 안내판의 이미지 이다.



기능:

- 주차장에 진입하는 입주민, 방문객에게 주차 현황을 알려 주차를 안내하는 전광판이다.
- 각 층별 실시간 주차 가능대수 또는 여유, 혼잡, 만차 상태를 실시간으로 표출한다.
만차=100%, 혼잡=60%, 여유=40%로 구분하여 표시
- 주차 가능 대수 및 주차 현황을 15 초 간격으로 번갈아 표시 해야 한다. (주기 설정)

컨트롤러 제어는 TCP 통신을 사용한다.

기본 TCP 포트 번호는 5000 이다.

2.4 단면 유도 전광판

단면 유도 전광판의 이미지 이다.



기능:

- 주차 공간 진입 시 차량 진입 방향을 유도한다.
- 해당 층의 주차 가능 대수 및 주차 현황(여유, 만차, 혼잡)을 표출한다.
만차=100%, 혼잡=60%, 여유=40%로 구분하여 표시
- 입주전에 “입주를 환영합니다” 문구를 제공한다.
- 주차 가능 대수 및 주차 현황을 15 초 간격으로 번갈아 표시 해야 한다. (주기 설정)

컨트롤러 제어는 TCP 통신을 사용한다.

기본 TCP 포트 번호는 5000 이다.

3. parkinfo package

3.1 type EndPoint

```
type EndPoint struct {
}
```

3.1.1 func NewClient(addr string) *EndPoint

- DESCRIPTION

주차 안내판 연결한다.

네트워크 환경에 따라 연결이 해지는 경우 자동으로 연결 시도하여 연결 상태를 유지한다.

- INPUT

addr string	주차 안내판 IP 주소 및 포트 정보 (예: "192.168.0.201:5000")
-------------	---

- RETURN VALUE

EndPoint	주차 안내판 생성 성공
nil	실패

- EXAMPLES

```
parkInfo := parkinfo.NewClient("192.168.0.201:5000")
```

3.1.2 func (*EndPoint) IsConnected() bool

- DESCRIPTION

주차 안내판 연결 여부를 확인한다.

- INPUT

없음

- RETURN VALUE

true	연결
false	연결 실패(시도 중)

- EXAMPLES

```
parkInfo := parkinfo.NewClient("192.168.0.201:5000")

if parkInfo.IsConnected() {
    fmt.Println("Conncted")
} else {
    fmt.Println("DisConncted")
}
```

3.1.3 func (*EndPoint) Close

- DESCRIPTION

주차 안내판 해제한다.

- INPUT

없음

- RETURN VALUE

없음

- EXAMPLES

```
parkInfo := parkinfo.NewClient("192.168.0.201:5000")
```

```
parkInfo.Close()
```

3.1.4 func (*EndPoint) Write (message Message) error

```

type InfoMsg struct {
    Text string          // 문자열
    Size int             // 3 (default) ~ 15
    Color string         // "white"(default), "red", "green", "yellow", "blue", "purple", "skyblue", "black"
    Effect string        // "default", (생략 가능), "shiftLeft", "shiftRight", "wipeLeft", "wipeRight",
                        // "blinkLeft", "blindRight", "blinkBG", "blinkText"
}

type Board struct {
    Format string          // "realtime", "page"
    Msg [] InfoMsg
}

```

- DESCRIPTION

주차 안내판에 메시지를 업데이트 한다.

- INPUT

type Message 안내판에 출력할 문자열

- RETURN VALUE

nil 메시지 갱신 성공
 Error() string 에러 문자열

- EXAMPLES

```

parkInfo := parkinfo.NewClient("192.168.0.201:5000")

// 1. realtime, 왼쪽: 혼잡, 오른쪽: 여유
parkInfo.Write(parkinfo.Board { Format: "realtime",
    Msg: []InfoMsg {{ Text: "혼잡", Color: "red" }, { Text: "여유", Color: "green" }}
})

parkInfo.Write(parkinfo.Board { Format: "page",

```

```
Msg: []InfoMsg {{ Text: “방문을 환영합니다.”, Size: 4, Color: “white”, Effect: “shiftRight” },  
})
```

4. 예제

```
p1 := parkinfo.NewClient("192.168.0.201:5000")
p2 := parkinfo.NewClient("192.168.0.202:5000")
p3 := parkinfo.NewClient("192.168.0.203:5000")
```

```
if (p1.IsConnected()) {
    p1.Write(parkinfo.Board { Format: "realtime",
        Msg: []InfoMsg {{ Text: "혼잡", Color: "yellow" }, { Text: "여유", Color: "green" }}
    })
}
```

```
if (p2.IsConnected()) {
    p2.Write(parkinfo.Board { Format: "realtime",
        Msg: []InfoMsg {{ Text: "여유", Color: "green" }, { Text: "만차", Color: "red" }}
    })
}
```

```
if (p3.IsConnected()) {
    p3.Write(parkinfo.Board { Format: "page",
        Msg: []InfoMsg {{ Text: "방문을 환영합니다.", Size: 4, Color: "white", Effect: "shfitLeft" }}
    })
}
```

5. 사용자 참고 사항

5.1 Known Bug

(진행 중)

5.2 개선 사항

6. 개발자 참고 사항

(추후 작성)