

vm_step1.dart

소스 코드

전체 코드

```
1 import 'package:buyoil/viewmodel/vm_serial_port.dart';
2 import 'package:flutter/cupertino.dart';
3 import 'package:riverpod_annotation/riverpod_annotation.dart';
4
5 import '../model/ui_state_step1.dart';
6
7 part 'vm_step1.g.dart';
8
9 @riverpod
10 class Step1 extends _Step1 {
11   @override
12   UIStateStep1 build() {
13     TextEditingController controller = TextEditingController();
14     controller.addListener(() {
15       state = state.copyWith(phoneNumber: controller.text);
16     });
17     return UIStateStep1.input(phoneNumber: "", controller: controller);
18   }
19
20   void pressedNumber(String buttonText) {
21     if(buttonText == '<') {
22       if (state.phoneNumber.isNotEmpty) {
23         String updatedPhoneNumber = state.phoneNumber.substring(0,
state.phoneNumber.length - 1);
24         state = state.copyWith(phoneNumber: updatedPhoneNumber);
25       }
26       return;
27     }
28
29     if(buttonText == 'v') {
30       ref.watch(serialPortVMProvider.notifier).sendPhoneNumber(state.phoneNumber);
31       // // 성공 시 다음 화면 이동
32       // state = UIStateStep1.completed(
33       //   controller: state.controller,
34       //   phoneNumber: state.phoneNumber,
35       // );
36       return;
37     }
38
39     if(state.phoneNumber.length < 11) {
40       state = state.copyWith(phoneNumber: state.phoneNumber + buttonText);
41       return;
42     }
43
44     print("pressedNumber: ${state.phoneNumber}");
```

```

45     }
46
47     void showErrorToast({int seconds = 2}) {
48         print("showErrorToast()");
49         state = state.copyWith(showErrorToast: true);
50         Future.delayed(Duration(milliseconds: seconds), () {
51             if(state.showErrorToast == true) {
52                 state = state.copyWith(showErrorToast: false);
53             }
54         });
55     }
56 }

```

1 파일 개요

클래스: `Step1 extends _$Step1`

역할:

- Step1 화면에서 **전화번호 입력 상태 관리**
- 사용자 입력(숫자 버튼/삭제/확인)을 처리
- Riverpod 상태 관리 (`@riverpod`)
- 입력 완료 시 **USB 포트 전송**과 연동 (`SerialPortVM.sendPhoneNumber`)

2 상태 정의

모델: `UIStateStep1`

- `phoneNumber` : 현재 입력된 전화번호 문자열
- `controller` : 입력 `TextEditingController`
- `showErrorToast` : 입력 오류/부적합 표시 여부

3 주요 기능

함수	설명
<code>build()</code>	초기 상태 생성 및 <code>TextEditingController</code> 리스너 등록
<code>pressedNumber(String buttonText)</code>	입력 버튼 처리:
	- < : 백스페이스 (마지막 문자 삭제)
	- v : 입력 완료 → <code>SerialPortVM.sendPhoneNumber()</code> 호출
	- 나머지 숫자 : 최대 11자리까지 입력 가능
<code>showErrorToast({int seconds = 2})</code>	오류 토스트 표시 → 지정 시간 후 자동 숨김

4 특징

1. 입력 시 즉시 `state.phoneNumber` 업데이트 → UI 자동 반영
 2. < 버튼으로 백스페이스 지원
 3. ✓ 버튼 클릭 시 `SerialPortVM`에 전화번호 전송 → USB 통신과 연결
 4. 토스트/에러 메시지 상태도 Riverpod 상태로 관리 가능
-

5 개선/확장 아이디어

- 숫자 입력 UI와 완전히 바인딩하여 키패드 버튼 클릭 → 상태 변화
- 입력 길이 제한, 유효성 검사 강화 (예: 010 시작 여부)
- USB 응답에 따라 화면 이동 처리 (Step2) → 이미 `SerialPortVM.listenByPort` 와 연동됨
- `showErrorToast` 를 `Fluttertoast/Scaffold` 메시지와 연동 가능