

# 로봇청소기(RVC) 제어 소프트웨어 요구사항명세서 (SRS)

> 표준: IEEE Std 830-1998 준수

구조적 분석(SA) 산출물 기반 통합 명세 대상: 로봇청소기(RVC) 내장 제어 소프트웨어

202411348 이시현

202111373 정일혁

202211285 김지태

## 1 서론 (Introduction)

### 1.1 목적 (Purpose)

본 문서는 로봇청소기 RVC 제어 소프트웨어의 기능적/비기능 요구사항을 명확히 기술하여, 설계·구현·시험의 기준과 상호 추적성을 제공한다. 특히 SA 결과에서 확인된 문제(Stop 상태 deadlock, Dust 미고려, 좌/우 회피 우선순위 부재, Backward 미고려, Motor-Cleaner 인터페이스 비일관성)를 해소한다.

### 1.2 범위 (Scope)

본 소프트웨어는 물체 회피 주행과 먼지 청소를 수행하며, 센서 인터페이스, 제어 로직(FSM), 액추에이터 인터페이스로 구성된다. 제품의 HW, 배터리·충전, 클라우드 연동, 지도 작성 등은 범위 밖이다.

### 1.3 정의/약어 (Definitions, Acronyms)

FSM: Finite State Machine

CN1: Motor Control FSM (이동/회피)

CN2: Cleaner Control FSM (청소/먼지처리)

Tick: 제어 주기(주기적 타이머 인터럽트)

Deadlock: 진전 없는 정지/교착 상태

DD: 데이터 사전(Data Dictionary)

액추에이터 : 외부 에너지를 받아 기계적인 움직임(회전 또는 직선)으로 변환하는 장치

### 1.4 참조 문서 (References)

IEEE Std 830-1998 (Software Requirements Specification)

(by 참고자료 : <https://dev-optimist.tistory.com/5>)

### 1.5 문서 개요 (Overview)

2장은 전체 기술개요, 3장은 구체 요구사항(기능/인터페이스/비기능), 4장은 데이터 사전, 5장은 상태기계, 6장은 검증, 7장은 추적성, 8장은 부록으로 구성한다.

## 2 전반적 기술 설명 (Overall Description)

### 2.1 제품 관점 (Product Perspective)

RVC는 외부환경(바닥, 장애물)과 상호작용한다. 소프트웨어는 센서로부터 논리상태를 취득해 제어로직(CN1, CN2)으로 판단하고, Motor/Brush/Vacuum 등의 액추에이터에 명령을 출력한다.

### 2.2 제품 기능 요약 (Product Functions)

장애물 감지 및 회피(좌/우 우선순위 규칙, 전방 막힘시 후진/탈출)  
먼지 감지 기반 청소 모드 관리(강/약, 집중청소, 일시정지 연동)  
데드록 감지 및 탈출(백워드/턴/재시도)  
안전 정지 및 오류 리포팅

### 2.3 사용자 특징 (User Characteristics)

일반 가정 사용자. 전원/시작, 모드 버튼, 기본 알림 인지 능력을 가정

### 2.4 제약사항 (Constraints)

제어 주기 50-100 ms 권장  
제한된 MCU 연산/메모리, 저전력 동작  
HW 센서 노이즈 및 지연 존재

### 2.5 가정 및 의존성 (Assumptions & Dependencies)

접촉/적외선/초음파 중 최소 1종 전방/좌/우 장애물 센서 제공  
Dust 센서(미세먼지/오염지표) 제공  
Motor/Brush/Vacuum 구동 가능, PWM/Direction 인터페이스 노출

## 3 특정 요구사항 (Specific Requirements)

### 3.1 외부 인터페이스 요구사항 (External Interface Requirements)

#### 3.1.1 센서 인터페이스

입력: Front\_Obs, Left\_Obs, Right\_Obs -> {0,1}, Dust\_Level -> [0..100],  
Bumper -> {0,1}  
타이밍: 모든 입력은 매 Tick 샘플링/필터링되어 Env\_State로 집계

#### 3.1.2 액추에이터 인터페이스

Motor: Motor\_Cmd -> {Forward, TurnLeft, TurnRight, Backward, Stop}  
Cleaner: Clean\_Cmd -> {Off, Normal, Boost, Spot}  
명령 유효시간: 1 Tick 이상 유지

#### 3.1.3 사용자/시스템 인터페이스

버튼: Start/Stop, Mode, Home(선택)  
알림: Buzzer/LED, 오류 코드 표시

### 3.2 기능 요구사항 (Functional Requirements)

FR-1 센서 전처리 및 상태산출

FR-1.1 Raw 센서값을 디바운스/이동평균으로 필터링한다.

FR-1.2 Env\_State = {Front, Left, Right, Dust\_Exist}를 생성한다. Dust\_Exist = (Dust\_Level  $\geq$   $\theta_{dust}$ ).

FR-2 제어노드 구조(CN1/CN2)와 일관성

FR-2.1 CN1(이동)과 CN2(청소)를 별도 FSM으로 분리한다.  
 FR-2.2 CN2는 청소 강화/집중 필요 시 Cleaner\_Trigger를 CN1에 전달하여 안전 정지/감속을 요구한다.  
 FR-2.3 CN1은 Trigger 수신 시 다음 Tick에서 Stop이 아닌 Pause 상태로 전이하고, Deadlock에 진입하지 않도록 한다.  
 FR-3 회피 우선순위 및 백워드 처리  
 FR-3.1 전방 장애물 시 좌/우 중 가용한 방향을 선택한다.  
 FR-3.2 좌/우 모두 가용 시 Left 우선 규칙을 적용한다(가변 파라미터화 turn\_priority=Left).  
 FR-3.3 좌/우 모두 불가 시 Backward를 T\_back 시간 수행 후 랜덤/우선 회전으로 탈출한다.  
 FR-4 데드록 감지/탈출  
 FR-4.1 k Tick 연속 위치 진전 없음 + 장애물 패턴 고착 시 Deadlock\_Suspect로 표식.  
 FR-4.2 Suspect 지속 시 Backward→Turn(Left)→Forward 시퀀스로 탈출 시도, N번 실패 시 사용자 알림.  
 FR-5 Dust 기반 청소 모드  
 FR-5.1 Dust\_Exist 시 Boost 또는 Spot 모드로 전환.  
 FR-5.2 일정 시간/영역 청소 후 Normal 복귀.  
 FR-5.3 CN1과 협조: Spot 동안 저속/정지 허용.  
 FR-6 오류/안전  
 FR-6.1 Bumper 동작 시 즉시 Backward 후 Pause.  
 FR-6.2 센서 Fail 검출 시 안전정지 및 코드 표출.

### 3.3 상태기계 요구사항 (FSM)

#### 3.3.1 CN1: Motor Control FSM

States: Idle, Forward, TurnLeft, TurnRight, Backwarding, Pause, Escape, Error  
 Initial: Idle  
 Transitions(핵심)  
 Idle → Forward (Start)  
 Forward → TurnLeft (Front && !Left && Right) 또는 (Front && Left && Right && priority=Left)  
 Forward → TurnRight (Front && Left && !Right)  
 Forward → Backwarding (Front && Left && Right)  
 → Pause (Cleaner\_Trigger || UserPause)  
 Backwarding → Escape ( $t \geq T\_back$ )  
 Escape → Forward (PathClear)

#### 3.3.2 CN2: Cleaner Control FSM

States: Off, Normal, Boost, Spot, Error  
 Initial: Off  
 Transitions  
 Off → Normal (Start)  
 Normal → Boost (Dust\_Exist && Dust\_Level  $\geq \theta\_boost$ )  
 Normal → Spot (Dust\_Exist && Localized)  
 Boost|Spot → Normal ( $t \geq T\_clean$  or Dust\_Clear)  
 → Error (SensorFault)

### 3.4 성능 요구사항 (Performance Requirements)

P-1 제어주기: 50–100 ms, 지터  $\pm 10$  ms 이내

P-2 회피 반응시간: 전방 감지→회피 명령까지  $\leq 150$  ms

P-3 Dust 감지→모드전환 ≤ 200 ms

### 3.5 신뢰성/안전/품질 (Non-Functional)

R-1 24h 연속 운전 중 크래시 0회, 회피 실패율 < 1% (테스트 코스)

M-1 파라미터( $\theta_{dust}$ ,  $T_{back}$ ,  $turn\_priority$ ) 설정 파일화

L-1 코드 규칙 MISRA-C 부분 준수 권고

## 4 데이터 사전 (Data Dictionary)

데이터명	타입	단위/범위	발생원	사용처	설명
Front_Obs	bool	{0,1}	전방센서	CN1	전방 장애물 유무
Left_Obs	bool	{0,1}	좌측센서	CN1	좌측 장애물 유무
Right_Obs	bool	{0,1}	우측센서	CN1	우측 장애물 유무
Dust_Level	int	0..100	Dust 센서	CN2	실시간 오염도
Dust_Exist	bool	{0,1}	전처리 CN2	임계 초과 여부	
Cleaner_Trigger	enum	{Pause, Slow}		CN2	CN1 청소 집중 시 이동 제어 요청
Motor_Cmd	enum	{Fwd, TL, TR, Bwd, Stop}		CN1	모터드라이브 최종 구동 명령
Clean_Cmd	enum	{Off, Normal, Boost, Spot}		CN2	클리너 청소 세기
Deadlock_Signal	bool	{0,1}	모니터	CN1	교착 의심 플래그
Tick	time	ms	타이머 전체	제어 주기 기준	

## 5 프로세스 명세 (Process Specifications)

### PS-1 Sensor Interface & Preprocessing

입력(Front/Left/Right, Dust\_Level)을 필터링(윈도우 N=3)하여 Env\_State와 Dust\_Exist를 산출.

### PS-2 Control Logic & Command Generation

CN1/CN2 두 FSM으로 구성. 상호 인터페이스는 Cleaner\_Trigger와 Motor\_Status.

데드록 검출기는 위치변화/명령변화/센서패턴을 k Tick 관찰하여 Deadlock\_Signal을 발생.

### PS-3 Actuator Interface

명령→PWM/Direction 변환. 유효성 점검(상호배제: Forward와 Backward 동시 금지).

## 6 검증 계획 (Verification)

요구사항 기반 테스트

TC-1 좌/우 둘 다 가능 시 좌회전 선택 확인 (FR-3.2)

TC-2 전면 막힘 시 후진  $T_{back}$  후 탈출 (FR-3.3)

TC-3 Dust 감지 시 Boost 전환 및 CN1 Pause 동작 (FR-5, FR-2.3)

TC-4 Deadlock 패턴 3회 반복 시 탈출 시퀀스 (FR-4)

TC-5 센서 Fail 시 안전정지 및 오류표시 (FR-6)

## 7 추적성 매트릭스 (Requirements Traceability)

Req ID	DFD/PS 매핑	FSM 매핑	테스트 케이스
FR-2.1	PS-2	CN1/CN2 분리	TC-3
FR-3.2	PS-2	CN1 Forward→TurnLeft	TC-1
FR-3.3	PS-2	CN1 Backwarding→Escape	TC-2
FR-4	PS-2	CN1 Escape 경로	TC-4
FR-5	PS-2	CN2 전이	TC-3
FR-6	PS-1/3	CN1/CN2 Error	TC-5

## 8 부록 (Appendices)

### 8.1 파라미터 표(초기값)

$\theta_{\text{dust}}=60$ ,  $\theta_{\text{boost}}=80$ ,  $T_{\text{back}}=600$  ms,  $\text{turn\_priority}=\text{Left}$

## 8.2 용어집(Glossary)

**Spot:** 국소 집중청소 모드

**Pause:** 안전 정지(Stop과 달리 Deadlock 회피 로직 포함)