

실리카 Lua 스크립트 작성 모범 사례 (Xilica Lua Scripting Best Practices)

패턴 명: InTable/OutTable 상태 추적 방식 (문자열 출력으로 변환)

이 문서는 Xilica Designer UI 컨트롤과 외부 TCP 브리지(Python)를 안정적으로 연결하기 위한 강력한 패턴을 설명합니다. 특히 다수의 채널을 효율적으로 관리하는 데 최적화되어 있습니다.

1. 핵심 개념

Lua 스크립트 내부에서 직접 TCP 소켓을 열려고 시도하는 대신(이는 불안정할 수 있음), Lua 스크립트를 상태 추적기 및 명령어 생성기로 사용합니다.

- 입력 (Inputs):** UI 값(페이더/버튼)이 InTable 을 통해 들어옵니다.
- 로직 (Logic):** Lua는 이전 실행 이후 입력값이 변경되었는지 확인합니다.
- 출력 (Outputs):** 변경이 감지되면 문자열 명령어(예: SET ch1v -10.0)를 포매팅하여 OutTable 로 보냅니다.
- 전송 (Transmission):** Xilica Designer의 외부 TCP 클라이언트 모듈이 이 문자열을 받아 브리지로 전송합니다.

2. 표준 코드 템플릿

```
-- =====
-- Xilica 상태 추적 템플릿 (Xilica State Tracking Template)
-- =====

local InTable = ...
if type(InTable) ~= "table" then InTable = {} end
local OutTable = {}

-- 1. 상태 저장소 초기화 (전역 변수)
-- 프레임 간 값을 기억하기 위해 전역 변수 'Previous_State'를 사용합니다.
if Previous_State == nil then
    Previous_State = {}
    for i=1, 10 do Previous_State[i] = -999.0 end -- 초기값으로 불가능한 값 사용
end

-- 2. 변경 사항 확인
local current_cmd = ""

for i = 1, 10 do
    local val = InTable[i]
    if val ~= nil then
        val = tonumber(val)

        -- 3. 변경 감지 로직
        -- (부동 소수점 오차를 피하기 위해 작은 임계값 사용)
        if math.abs(val - Previous_State[i]) > 0.001 then

            -- 4. 명령어 문자열 생성
            current_cmd = string.format("SET param_%d %.1f\r", i, val)

            -- 5. 상태 업데이트
```

```

        Previous_State[i] = val

        -- 6. 루프 중단 (네트워크 폭주를 막기 위해 한 사이클당 하나의 명령만 전송)
        break
    end
end
end

-- 7. TCP 블록으로 출력
if current_cmd ~= "" then
    OutTable[1] = current_cmd
else
    OutTable[1] = ""
end

return OutTable, ""

```

3. 장점

1. **안정성**: 소켓 오류로 인해 Lua 스크립트가 중단될 위험이 '0'에 가깝습니다.
2. **깔끔한 배선**: 수많은 와이어를 하나의 문자열(String) 출력으로 묶어서 처리합니다.
3. **효율성**: `break` 를 사용하여 출력을 자연스럽게 조절(Rate-limiting)하므로, 네트워크 폭주를 방지합니다.

4. Designer 배선 방법

1. **Lua 스크립트 모듈**: 이 스크립트를 로드하고, 제어할 채널 수만큼 입력 핀(Input Pins)을 설정합니다.
2. **TCP 클라이언트 모듈**: Lua 모듈의 `Output 1` (String)을 TCP 클라이언트의 입력에 연결합니다.
3. **포트 구성**: TCP 클라이언트의 원격 포트(Remote Port)를 `1500` (또는 사용 중인 브리지 포트)으로 설정합니다.