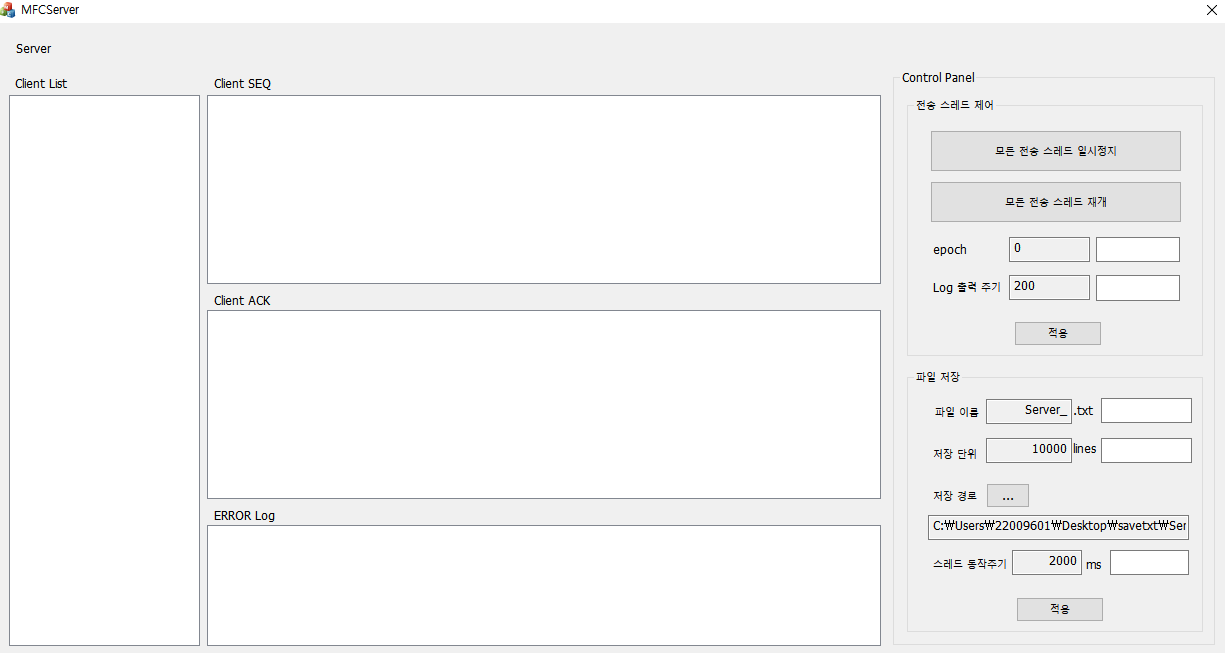
**목차**

1. **화면 구성**
   1. **메인 화면**
   2. **Control Panel**
2. **파라미터 설정**
   1. **전송 스레드 파라미터 설정**
   2. **파일 저장 파라미터 설정**
3. **서버 - 클라이언트 연결 및 프로그램 동작**
   1. **클라이언트 생성**
   2. **메시지 규칙**
   3. **에러 로그**
   4. **파일 저장**

**1. 화면 구성**

**1.1 메인 화면**

서버, 클라이언트 프로그램을 실행하면 다음과 같은 창이 생성된다. (클라이언트는 클라이언트 생성 버튼이 추가적으로 있다.)



**5**

**4**

**3**

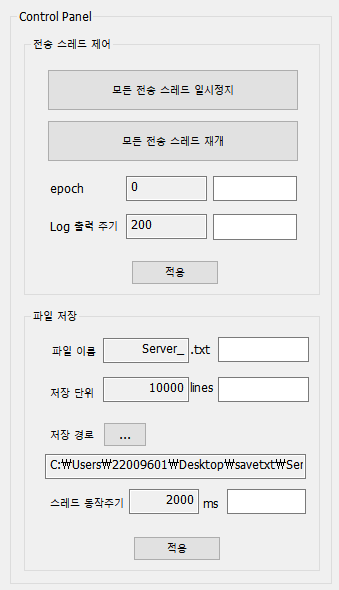
**2**

**1**

1. Client List : 연결된 클라이언트의 리스트를 표시한다.
2. Client SEQ : 연결된 클라이언트로부터 받은 SEQ 로그 데이터를 표시한다.
3. Client ACK : 연결된 클라이언트로부터 받은 응답(ACK) 로그 데이터를 표시한다.
4. Error Log : 송수신 과정에서 발생하는 모든 에러 로그를 표시한다.
5. Control Panel : 파라미터 설정 및 스레드를 제어한다.

**1.2 Control Panel**

Control Panel은 다음과 같이 구성된다.



**10**

**9**

**8**

**5**

**6**

**7**

**4**

**3**

**2**

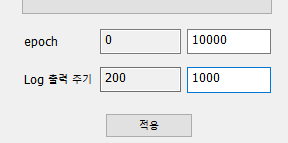
**1**

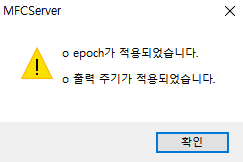
1. 모든 전송 스레드 일시정지 버튼 : 서버의 모든 전송 스레드를 일시정지한다.
2. 모든 전송 스레드 재개 버튼 : 서버의 모든 전송 스레드를 재개한다.
3. Epoch : 전송 스레드가 몇 번 전송하는지에 대한 값을 입력한다. Epoch만큼의 데이터를 전송하고 전송을 중지하며 스레드 재개 버튼을 누르면 다시 epoch만큼 전송하고 중지한다. 기본값은 0이고 값이 0이면 무한히 전송한다.
4. Log 출력 주기 : 로그 출력 화면에 몇 번의 주기로 로그를 출력할 지에 대한 값을 입력한다. 기본값은 200이다.
5. 적용 버튼 : 전송 스레드 제어에 대한 모든 값을 설정하는 버튼이다.
6. 파일 이름 : 저장할 파일이름을 입력한다. 기본값은 Server\_, Client\_ 이다.
7. 저장 단위 : 1개의 파일에 저장될 데이터의 양에 대한 값을 입력한다. 기본값은 10000이다.
8. 저장 경로 : 저장 경로를 설정하는 버튼이다. 클릭 시 파일 다이얼로그가 생성된다.
9. 스레드 동작 주기 : 저장을 담당하는 스레드가 동작하는 주기를 입력한다. 기본값은 2000ms 이다.
10. 적용 버튼 : 파일 저장에 대한 모든 값을 설정하는 버튼이다.

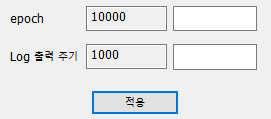
**2. 파라미터 설정**

**2.1 전송 스레드 파라미터 설정**

Epoch와 Log 출력 주기를 입력하고 적용 버튼을 누르면 각 파라미터 값이 적용된다.

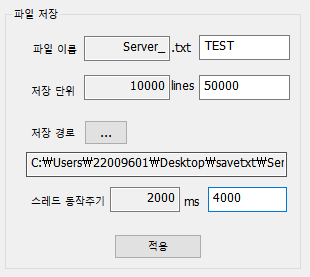




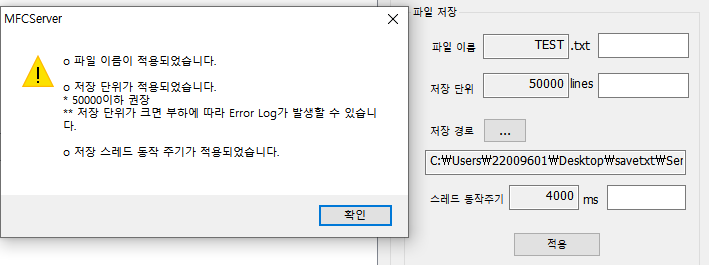


**2.2 파일 저장 파라미터 설정**

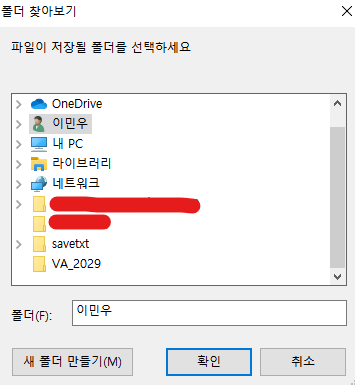
파일 이름, 저장 단위, 스레드 동작 주기를 설정하고 적용 버튼을 눌러 각 파라미터 값을 적용한다.



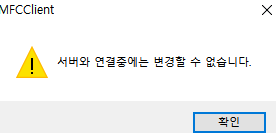
저장 단위는 1개의 text파일에 저장될 라인 수를 설정한다. 저장 단위가 클수록 저장 스레드에 걸리는 부하가 커지기 때문에 상황에 따라 에러 로그가 발생할 수 있다. 따라서 2000~100000 lines단위로 범위를 제한하였다. 스레드 동작 주기는 저장 스레드가 깨어나 일을 하는 주기를 설정한다. 스레드 동작 주기가 작으면 그 만큼 자주 깨어나 저장 시퀀스를 수행하지만 부하가 더 발생될 수 있고, 크면 저장이 지연될 수 있다.



각 파라미터는 필수 입력사항이 아니며 입력이 없을 시 기본값으로 설정된다.



저장 경로 설정은 [ **…** ]버튼을 클릭하여 파일 다이얼로그에서 경로를 선택한다. 새 폴더를 만들어 경로를 만들 수 있고 원하는 경로를 클릭한 뒤 확인버튼을 눌러 설정한다.

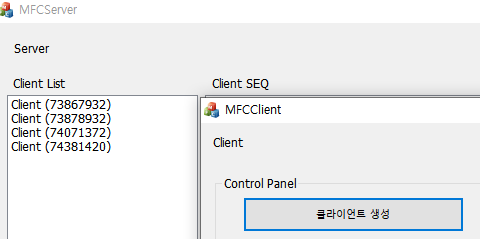


Log 출력 주기를 제외한 모든 파라미터는 연결 중에는 변경할 수 없으므로 서버와 클라이언트의 파라미터 값을 모두 설정한 뒤 서버와 클라이언트를 연결한다.

**3. 서버-클라이언트 연결 및 동작**

**3.1 클라이언트 생성**

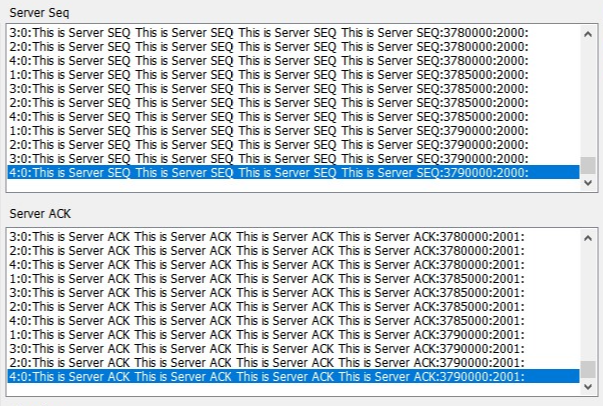
클라이언트 프로그램에서 원하는 클라이언트 수만큼 클라이언트 생성 버튼을 클릭한다.



클라이언트가 생성되면 서버와 클라이언트의 클라이언트 리스트에 연결된 클라이언트 리스트가 표시되고 바로 송수신을 시작한다.

**3.2 메시지 규칙**

다음은 클라이언트의 수신 로그이다.



송수신 메시지의 규칙은 ‘:’ 구분자를 이용하여 각 필드를 구분한다.

메시지 규칙은 다음과 같다.













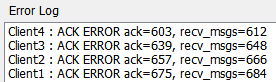
클라이언트 번호 **:** Flag **:** Text 데이터 **:** SEQ 또는 ACK번호 **:** 식별번호 **:**

**<식별번호>**

|  |  |  |
| --- | --- | --- |
|  | SEQ | ACK |
| 서버 | 2000 | 2001 |
| 클라이언트 | 1000 | 1001 |

**3.3 에러 로그 분석**

에러 로그는 수신된 데이터에 대해 오류가 생긴 데이터의 로그를 보여준다.









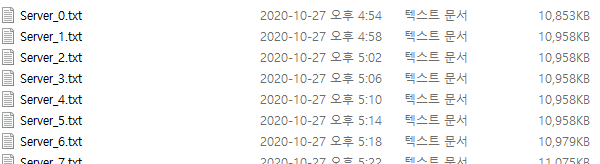
에러 로그에서 몇 번 클라이언트에서 오류가 발생했는지 정보가 맨 앞에 표시되고, ACK에러는 ACK ERROR, SEQ에러는 SEQ ERROR로 표시된다.

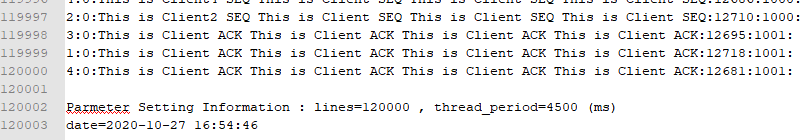
에러의 범위는 ack=정수1 또는 seq=정수1, recv\_msgs=정수2 형태로 표시되며 (정수1)~(정수2 - 1)까지 이다.

예를 들어, 위의 에러 로그의 경우 ACK 오류이며 ACK번호 603번부터 611번까지 문제가 생겼다는 것을 알 수 있다.

**3.4 파일 저장**

정상적으로 수신한 데이터와 설정한 파라미터 정보, 시간 정보와 함께 기록되어 설정 경로에 txt포맷 파일로 저장된다. 에러가 생긴 데이터에 대해서는 저장되지 않는다.





프로그램 구조도

