



CODESCROLL
CONTROLLER TESTER



CODESCROLL **CONTROLLER TESTER** **v3.0.8 Release Note**

Software for safe world

www.suresofttech.com

CODESCROLL™Controller Tester Release Note

문서정보

문서 버전: 3.0.8

작성일: 2019년 05월 13일

배포일: 2019년 05월 16일

기술지원

support@suresofttech.com

+82-2-6472-2800

최종고지

Copyright © Suresoft Technologies, Inc., All rights reserved.

Proprietary and Confidential

This material is proprietary to Suresoft Technologies, Inc. It contains trade secret and confidential information which is solely the property of Suresoft Technologies, Inc.

This material is for client's internal use only. This material shall not be used, reproduced, copied, disclosed, and transmitted, in whole or in part, without the express consent of Suresoft Technologies, Inc.

슈어소프트테크㈜ 허락 없이 무단 전재 및 배포하는 것을 금합니다.

본 문서 또는 제품에 대한 제안, 문의 사항은 슈어소프트테크㈜로 연락해주시시오.

이 릴리즈에는 새로운 기능 및 버그 수정 사항이 포함되어 있습니다.

목차

1. 추가된 기능	4
타깃 테스트 결과 자동 업데이트	4
타깃 코드 내보내기 시 빌드 스크립트 실행	6
함수 커버리지 지원	6
2. 기능 개선	9
Green Hills MULTI IDE 프로젝트(*.gpj) 가져오기 기능 개선	9
대용량 테스트 성능 개선	9
3. 버그 수정	10
버그 수정 목록	10

1. 추가된 기능

타깃 테스트 결과 자동 업데이트

Controller Tester는 타깃 로그 수집기를 이용하여 타깃에서 수행한 테스트 결과를 자동으로 업데이트할 수 있습니다.

타깃 로그 수집기

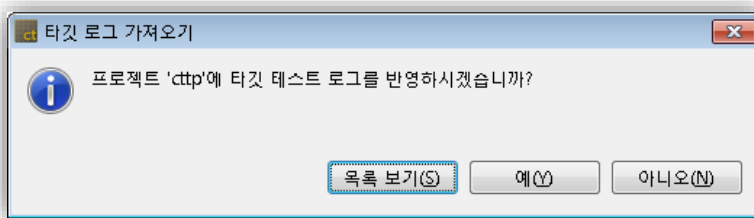
타깃 로그 수집기는 타깃 테스트 결과를 수집하여 Controller Tester로 전송합니다. 타깃 테스트 결과 수집 방법은 크게 통신과 파일 스캔으로 나뉘며, 지원하는 통신 프로토콜과 스캔 가능한 파일 형식은 다음과 같습니다.

통신 프로토콜	TCP, UDP, UART(Serial)
파일 형식	타깃 로그(text), 메모리 덤프(Hex, Intel HEX)

설치, 실행 방법은 CT_Target_Plugin_Manual.docx의 [타깃 로그 수집기]를 참고 바랍니다.

타깃 테스트 로그 자동 감지

Controller Tester는 타깃 로그 수집기와 주기적으로 통신하여 타깃 로그 수집기에 추가된 타깃 테스트 로그를 자동으로 감지합니다. 자동 감지 주기는 [환경 설정]-[타깃 로그 수집기]에서 설정할 수 있습니다.



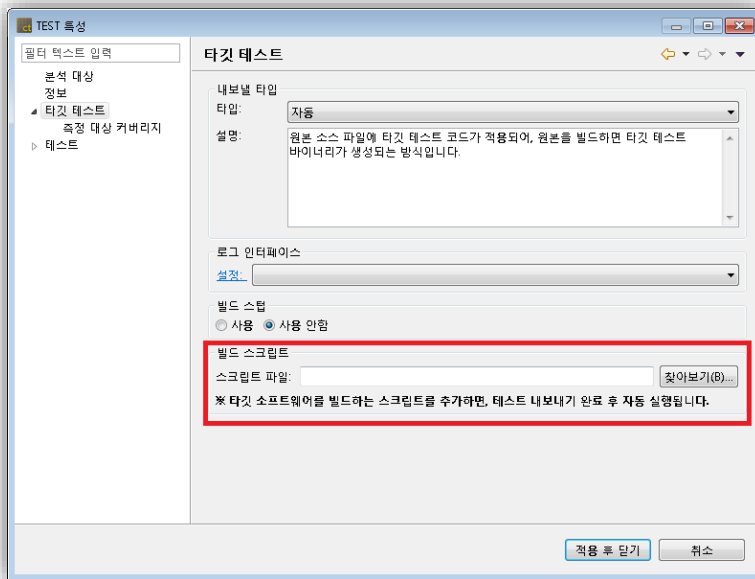
타깃 테스트 로그 가져오기 방식 추가

타깃 테스트 로그 가져오기 방식에 [압축 파일로부터 가져오기]와 [타깃 로그 수집기로부터 가져오기]가 추가되었습니다.

방식	설명
압축 파일로부터 가져오기	타깃 로그 수집기가 생성한 타깃 로그 압축파일로부터 타깃 테스트 결과를 가져옴
타깃 로그 수집기로부터 가져오기	타깃 로그 수집기로부터 최신 타깃 테스트 결과를 가져옴

타겟 코드 내보내기 시 빌드 스크립트 실행

타겟 소프트웨어를 빌드하는 스크립트를 추가하면, 테스트 내보내기 완료 후에 빌드 스크립트가 자동 실행됩니다.



함수 커버리지 지원

Controller Tester는 소프트웨어 통합시험에 필요한 함수 커버리지를 지원합니다.

※ ISO 26262 소프트웨어 파트(Part 6)의 Table 15

함수 커버리지는 커버리지 뷰, 테스트 뷰, 테스트 보고서에서 제공됩니다. 각 함수별 수행 여부는 Y/N으로 표현하며, 프로젝트에 대한 함수 커버리지는 전체 함수 중 한번이라도 호출된 함수의 수를 백분율로 표현합니다.

커버리지 뷰

커버리지 뷰에 '함수' 컬럼이 추가되었습니다.

※ 프로젝트에 대한 함수 커버리지는 [전체 커버리지 보기]와 [전체 커버리지 보기(외부 커버리지)] 기능을 선택한 경우에만 표시됩니다.

커버리지: 오류

zlib 프로젝트의 호스트 커버리지 정보를 보여줍니다.

	대상 함수	구문	분기	MC/DC	함수 호출	함수
1	tr_align(struct internal_state *)	71.42% (15/21)	75.00% (3/4)	50.00% (1/2)	100.00% (1/1)	Y
2	tr_flush_bits(struct internal_state *)	100.00% (1/1)	N/A	N/A	100.00% (1/1)	Y
3	tr_flush_block(struct internal_state *, char *, unsigned lo...	57.14% (24/42)	68.75% (11/16)	50.00% (5/10)	80.00% (8/10)	Y
4	tr_init(struct internal_state *)	100.00% (10/10)	N/A	N/A	100.00% (2/2)	Y
5	tr_stored_block(struct internal_state *, char *, unsigned l...	100.00% (11/11)	100.00% (2/2)	100.00% (1/1)	100.00% (1/1)	Y
6	tr_tally(struct internal_state *, unsigned int, unsigned int...	22.22% (2/9)	0.00% (0/4)	0.00% (0/2)	N/A	Y
7	adler32(unsigned long, const unsigned char *, unsigned...	98.05% (101/103)	87.50% (21/24)	75.00% (9/12)	N/A	Y
8	adler32_combine(unsigned long, unsigned long, signed...	100.00% (1/1)	N/A	N/A	100.00% (1/1)	Y
9	adler32_combine64(unsigned long, unsigned long, sign...	100.00% (1/1)	N/A	N/A	100.00% (1/1)	Y
10	adler32_combine(unsigned long, unsigned long, signed...	100.00% (21/21)	100.00% (10/10)	100.00% (5/5)	N/A	Y
11	bi_flush(struct internal_state *)	100.00% (9/9)	100.00% (4/4)	100.00% (2/2)	N/A	Y
12	bi_reverse(unsigned int, signed int)	100.00% (5/5)	100.00% (2/2)	100.00% (1/1)	N/A	Y
합계		43.88% (1962/...	30.08% (723/2...	18.53% (265/1...	44.75% (175/3...	100.00% ...

타겟 환경에서는 Asm 코드가 포함된 함수의 커버리지를 측정할 수 없습니다.

테스트 뷰

테스트 뷰에 함수 커버리지가 추가되었습니다. 함수 커버리지 선택 시, 테스트 뷰의 대시보드에는 해당 프로젝트의 함수 커버리지가 표시됩니다.

유닛 테스트 뷰의 경우, 함수 노드에 함수의 수행 여부가 표현됩니다.

유닛 테스트 통합 테스트

실행

함수 커버리지

(743 / 0 / 185) 928 97.0% (132/136)

파일, 함수, 테스트, 상태, 이슈 입력

이름	결과	커버리지
> adler32 (unsigned long, const unsigned char *...)	(6 / 0 / 7) 13	Y
> adler32_combine (unsigned long, unsigned long...)	(5 / 0 / 0) 5	Y
> adler32_combine64 (unsigned long, unsigned long...)	(5 / 0 / 0) 5	Y
> adler32_combine_ (unsigned long, unsigned long...)	(11 / 0 / 0) 11	Y
> compress (unsigned char *, unsigned long *...)	(3 / 0 / 0) 3	Y
> compress2 (unsigned char *, unsigned long *...)	(6 / 0 / 0) 6	Y
> compressBound (unsigned long)	(3 / 0 / 0) 3	Y
> crc32 (unsigned long, const unsigned char *...)	(6 / 0 / 1) 7	Y
> crc32_big (unsigned long, const unsigned char *...)	(9 / 0 / 1) 10	Y
> crc32_combine (unsigned long, unsigned long...)	(5 / 0 / 0) 5	Y
> crc32_combine64 (unsigned long, unsigned long...)	(5 / 0 / 0) 5	Y
> crc32_combine_ (unsigned long, unsigned long...)	(9 / 0 / 0) 9	Y
> crc32_little (unsigned long, const unsigned char *...)	(9 / 0 / 1) 10	Y
> deflate (struct z_stream_s *, signed int)	(30 / 0 / 5) 35	Y
> deflateBound (struct z_stream_s *, unsigned long)	(10 / 0 / 0) 10	Y
> deflateCopy (struct z_stream_s *, struct z_stream_s *)	(0 / 0 / 3) 3	Y
> deflateEnd (struct z_stream_s *)	(19 / 0 / 7) 26	Y
> deflateInit2_ (struct z_stream_s *, signed int...)	(20 / 0 / 0) 20	Y
> deflateInit (struct z_stream_s *, signed int...)	(5 / 0 / 0) 5	Y
> deflateParams (struct z_stream_s *, signed int...)	(15 / 0 / 0) 15	Y
> deflatePending (struct z_stream_s *, unsigned int *...)	(5 / 0 / 0) 5	Y
> deflatePrime (struct z_stream_s *, signed int...)	(7 / 0 / 0) 7	Y
> deflateReset (struct z_stream_s *)	(1 / 0 / 0) 1	Y
> deflateResetKeep (struct z_stream_s *)	(5 / 0 / 0) 5	Y

테스트 보고서

'전체 커버리지'와 '함수별 커버리지'에 '함수' 컬럼이 추가되었습니다. 각각에서 프로젝트에 대한 함수 커버리지와 함수의 수행 여부를 확인할 수 있습니다.

2. 기능 개선

Green Hills MULTI IDE 프로젝트(*.gpj) 가져오기 기능 개선

1. 매크로 지원
2. 소스 파일별 컴파일 플래그 지원
3. srcDir 옵션 처리
4. 가상 드라이브 구성 지원

대용량 테스트 성능 개선

1. 함수 이름의 길이가 길 때, 테스트 편집기 성능 개선
2. 테스트 정보의 메모리 사용량 1/4로 줄임
3. 테스트 데이터 로드 쿼리 성능 개선
대시보드에서 프로젝트 상태 확인 방법 추가

3. 버그 수정

버그 수정 목록

1. CT RTV 타깃 출력값 데이터 내보내기(csv) 시, 정상적으로 출력값이 내보내지 못하는 문제 수정
2. 이름이 __S로 끝나는 변수 또는 함수에 대해 스텝이 생성되지 않는 문제 수정
3. main 함수가 스텝으로 생성되지 않는 문제 수정
4. 시그니처가 동일한 함수가 존재 시, 타깃 로그 가져오기가 실패하는 문제 수정
5. 타깃 테스트 로그 가져오기 수행 시 커버리지가 누적되지 않는 문제 수정
6. 테스트 내보내기 시 파일 정보를 같이 포함하여 내보내도록 수정
7. 테스트 가져오기 시 동일한 이름의 함수가 있는 경우, 상대 경로와 파일명 일치 여부를 확인하여 테스트를 매칭하도록 수정
8. 변수 이름이 __C로 끝나는 경우 스텝 생성이 안되는 문제 수정
9. 함수 시그니처가 같고 다른 파일에 정의되어 있는 경우 커버리지 가져오기 정상적으로 되지 않는 문제 수정
10. 분석 시 DB 파일의 크기가 비정상적으로 증가하는 문제 수정
11. 함수 정의 소스 파일과 함수 호출 소스파일이 다른 경우, 소스 편집기 창에서 '스텝 생성' 컨텍스트 메뉴가 비활성화되는 문제 수정
12. 테스트 별 보고서의 데이터가 65536 라인 이상인 경우, 데이터가 누락되는 문제 수정
13. WHITE(0), BLACK(1)과 같이 테스트 데이터에 enumerator에서 추가한 이름이 있는 경우 발생하는 문제 수정

-
14. return 값이 char * 타입이고 테스트 대상 함수가 null을 반환하는 경우 ucli.exe 죽는 문제 수정
 15. 내보내기 한 데이터의 파일 경로를 DB에서 찾지 못하여 발생하는 문제 수정
 16. 통합 테스트 보고서에 통합 테스트의 설명이 누락되는 문제 수정
 17. 테스트 정보 탭에서 static 변수 관련하여 작업 시, NullPointerException이 발생하는 문제 수정
 18. 테스트 정보 편집 영역에서 마지막으로 편집한 상태가 저장되지 않는 문제 수정
 19. CT RTV 커버리지 색 표시 건 문제 수정
 20. 테스트 편집기 창에서 입력 출력 체크 박스 풀림 문제 수정
 21. asm 처리 옵션을 사용할 때 발생하는 커버리지 탐침 문제 수정
 22. 테스트 뷰의 검색 및 필터 창에서 소스코드의 키워드 붙여넣기 기능이 오동작한 문제 수정
 23. 동일한 테스트의 테스트 케이스 선택 변경 시, 변경한 enum 입력값과 기대값이 변경 전 상태로 설정되는 문제 수정
 24. enum 타입의 기대값 수정 시 기대값이 사라지는 문제 수정
 25. CT RTV 프로젝트 설정 창 문제 수정
 26. vs 프로젝트 [소스코드 복원] 기능 수행 시, 정상적으로 프로젝트 복원이 안되는 문제 수정
 27. 매크로 상수에 정의된 리터럴 접미사 관련 문제 수정
 28. 16진수 변환 시 Exception 발생하는 문제 수정
 29. 테스트 케이스 창에서 NullPointerException 문제 수정
 30. 테스트 가져오기 기능에서 덮어쓰기 수행 시, 테스트를 덮어쓰지 않는 문제 수정
 31. Visual Studio 프로젝트 삭제된 파일이 업로드 데이터에 포함되는 문제 수정
 32. 소스 경로가 다른 동일한 시그니처의 inline 함수 커버리지 계산 처리 시 ucli.exe 중지되는 문제 수정
 33. CLI에서 xlsx 확장자의 보고서가 생성 안되는 문제 수정
 34. 비관리 코드일 때 진법을 변경할 수 없는 문제 수정
-