



<1차 저장>

```
HOOK(Script.dll!0x0001C212,TRANS(EDX,TWOBYTE,OVERWRITE(IGNORE)),RETNPOS(COPY)),FORCEFONT(5),FONT(AralGothic,-13),ENCODEKOR
```

HOOK(Script.....,TRANS(EDX,TWOBYTE,OVERWRITE(IGNORE))

위치는 이전기록을 참조하면 이해가 쉽다. EDX를 메모리 덮어쓰기로, 버퍼크기는 무시하며, 1바이트를 강제로 2바이트 문자로 변환한다.

**FONT(AralGothic,-13)**

아랄고딕 폰트를 사용하고 폰트는 아랄이 지정해주는기본 크기로 하였다.

## ENCODEKOR

EUC-KR를 SHIFT-JIS로 변환하는 설정값

이외의 설정을 건드리면 높은확률로 크래시가 나니까 조심하 건드려봐야한다.

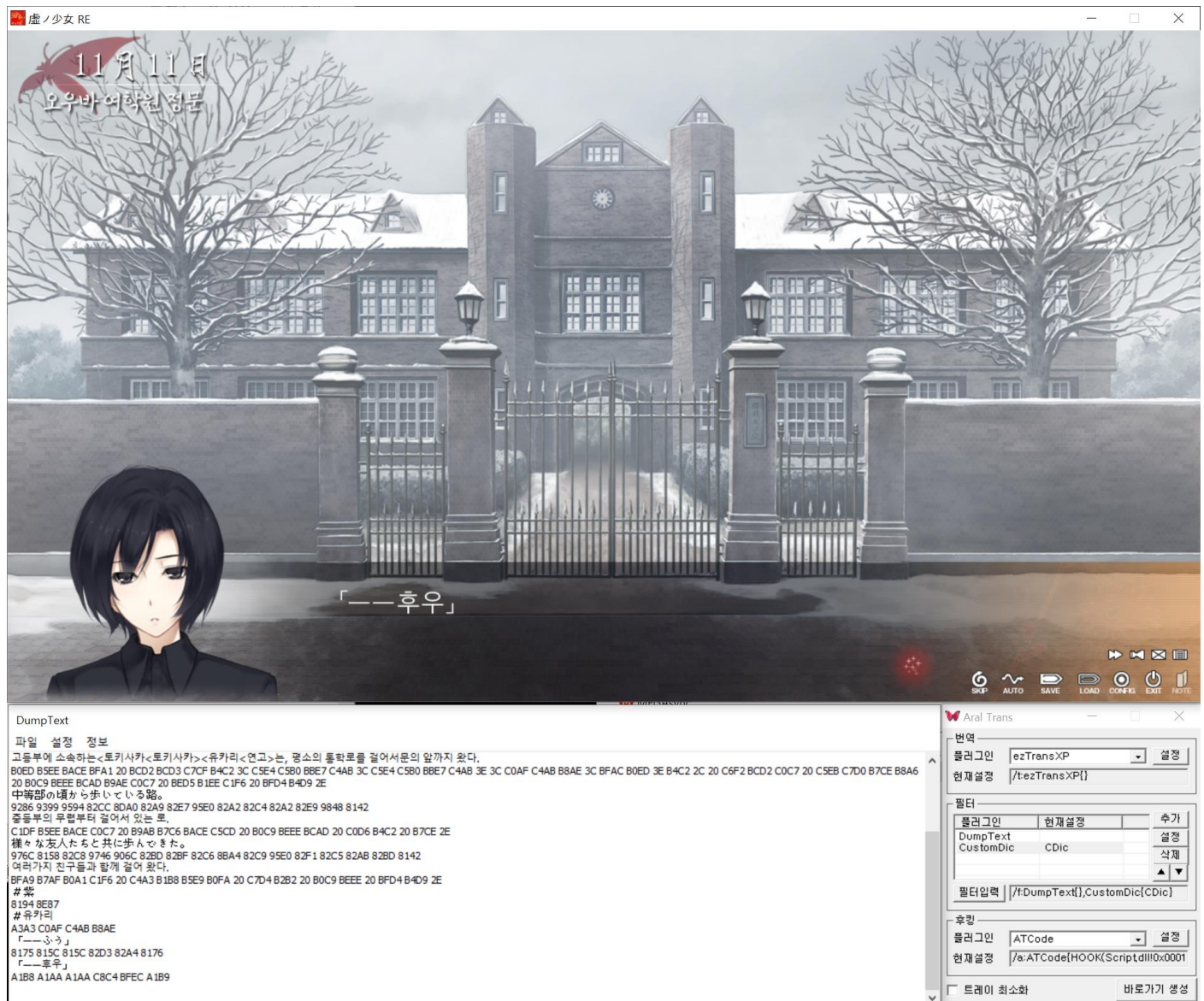
일단 대사 번역은 되지만

이름, 대사가 함께출력되지 않고 두번에 나뉘어서 출력되는 상황이다.





#유카리..



원래라면

유카리 「——후우」

이렇게 출력되어야 하는부분.

FixLine 옵션으로 해결이 된다는것같은데 더 알아봐야겠다.

SAVE

データをセーブします

1

2

3

4

5

6

7


8

9

10

EXIT

FILE.001



DAY


2023/04/15 21:39:11

SCENE

序詞 11夕11日

昭和三十二年、十一月。

FILE.002



DAY


2023/04/18 16:53:35

SCENE

序詞 11夕11日

武蔵岡の大地で明治期より良の

FILE.003



DAY

2023/04/18 18:24:22


SCENE

序詞 11夕11日

#紫

最新

FILE.004



DAY

2023/04/18 18:24:33

SCENE

序詞 11夕11日

#小羽

FILE.005

DAY

SCENE

FILE.006

DAY

SCENE

FILE.007

DAY

SCENE

FILE.008

DAY

SCENE

FILE.009

DAY

SCENE

FILE.010

DAY

SCENE

# DumpText

## 파일 설정 정보

```

89BD 82F0 9862 82B9 82CE 82A2 82A2 82CC 82BE 82EB 82A4 8142
두엇을 이야기하면 좋겠지.
89AB BEF9 C0B8 20 C0CC BEDF B1E2 C7CF B8E9 20 C1C1 B0DA C1F6 2E
何処から来たのかを聞けばいいのだろうか。
89BD 8F88 82A9 82E7 9788 82BD 82CC 82A9 82F0 95B7 82AF 82CE 82A2 82A2 82CC 82BE 82EB 82A4 82A9 8142
어딘가에서 왔는지를 물으면 좋은 것일까.
BEEE B5F2 B0A1 BFA1 BCAD 20 BFD4 B4C2 C1F6 B8A6 20 B9B0 C0B8 B8E9 20 C1C1 C0BA 20 B0CD C0CF B1EE 2E
そんな事を考えながら自分の席に腰を下ろす。
82B8 82F1 82C8 8E96 82F0 8D6C 82A6 82C8 82AA 82E7 8EA9 95AA 82CC 90C8 82C9 8D98 82F0 89BA 82EB 82B7 8142
그런 일을 생각 하면서 자신의 자리에 앉는다.
B1D7 B7B1 20 C0CF C0B8 20 B8FD B0A2 20 C7CF B8E9 BCAD 20 C0DA BDC5 C0C7 20 C0DA B8AE BFA1 20 BEC9 B4C2 B4D9 2E
教室内で転入生の話を持ちきりなのか、珍しくぞわついていた。
88B3 8EBA 93E0 82CD 93FD 93FC 90B6 82CC 9862 82C5 8E9D 82BF 82AB 82E8 82C8 82CC 82A9 8141 92BF 82B5 82AD 82B4 82ED 82C2 82A2 82C4 82A2 82BD 8142
교실내는 전입생의 이야기로 가득 분인가, 드물게 웅성거리고 있었다.
B1B3 BDC7 B3B8 B4C2 20 C0FC C0D4 B8FD C0C7 20 C0CC BEDF B1E2 B7CE 20 B0A1 C1AE 20 BBD3 C0CE B0A1 2C 20 B5E5 B9B0 B0D4 20 BFF5 BCBA B0C5 B8AE B0ED 20 C0D6 BEFA B4D9 2E
#小羽
8194 8FAC 8948
#크리네
A3A3 C4DA C7CF B3D7

```

번역

플러그인

ezTransXP

설정

현재설정

/tezTransXP{}

필터

플러그인	현재설정	추가
DumpText		설정
CustomDic	CDic	삭제

필터입력

/fDumpText{},CustomDic[CDic]

후킹

플러그인

ATCode

설정

현재설정

/s:ATCode[HOOK(Script.dll)0x0001]

트레이 최소화

바로가기 생성

로그가 언제 어떻게 기록되는지를 알아봐야한다.

둘째에 복사된 것은 다목적(?)임시저장소, 셋째는 출력을 위한 임시저장소?

코드파인딩기록을 보고 셋째와 로그기록 메모리간의 관계를 다시 확인해보자..

해야할것이 두세배로 늘어난듯한...



이번 기록은 알맹이가 없다.

인코딩문제는 AT코드 옵션으로 아주 간단히 해결됐고, 글자수 맞추기도 완료했다.

그러나 "공허의소녀"에서 문장이 들어가는 부분은 대사뿐만이 아니라 수첩이라는 것이 있는 듯하다.

이것이 이미지번역으로 해결가능한지는.. 플레이 해보지 않은 입장에서는 알수가 없으니 알아봐야할 과제중 하나이다. 아마 문자열 받아서 주는거같긴한데 이미지번역이면 좋겠구나...

그리고 또다른 문자열 사용처가 어디있는지 더 찾아봐야할지도 모른다....



해야할것만 정리하고 마무리하자.

1. 로그를 어느 메모리에 기록하는지 찾아내기. (option. 로그를 기록하면서 글자수와 관련된 것이 있는가?)
2. 로그로 복사하는 원본(첫째의 원본과는 다른, parent 개념..?)이 어디서 온것인가를 찾는다면 그것을 메모리 덮어쓰기로 바꿀경우, 로그도 한글로 기록이 되지 않을까?
3. 수첩이 어디에 있는지 찾아내기.

진짜 마지막으로 뇌피셜로 추측만해봄.

최초의 원본문장은 첫째에 존재함.

둘째는 임시저장소로 사용되며 이 공간으로부터 로그, 대사창으로 문자열이 전달된다.

셋째는 대사창으로 출력하기 직전 임시 저장소로 사용된다. 이들은 Himorogi의 BufferUpdater? FrameBuffer?(이름이 기억안남)로 전달되며 이쪽에서 화면 출력이 이루어지는듯하다.

