

산업용 플랜트 접지시스템의 전위분포 해석기법에 관한 연구

1994. 12. 31



한국산업안전공단
KOREA INDUSTRIAL SAFETY CORPORATION
산업안전연구원
INDUSTRIAL SAFETY RESEARCH INSTITUTE

제 출 문

한국산업안전공단 이사장 귀하

본 보고서를 산업안전 연구개발 및 보급 사업의 일환으로 수행
한 “산업용 플랜트 접지시스템의 전위분포 해석기법에 관한 연구”
사업의 최종 보고서로 제출합니다.

1994. 12. 31

주관연구부서 : 산업안전연구원
기계전기연구실

연구수행자 : 선임연구원 이 형 수

머 리 말

접지란 전기, 전자, 통신 분야에서 안전을 보장하여 주는 가장 유효한 수단으로서 이를 잘 시행하면 전기안전의 절반을 달성할 수 있다고 할 수 있을 만큼 기본적인 기술입니다. 그러나, 이러한 중요성에도 불구하고 이 분야의 기술이 소홀히 다루어져 왔습니다. 이는 전기안전을 도모하는 데 있어 대단히 우려할 일이 아닐 수 없습니다. 더구나, 산업사회가 고도정보화 사회로 이행되면서 지금까지 문제시하지 않았던 전자적인 환경에 민감한 전자기기에 대한 접지측면의 적절한 대책이 수립되지 않으면 많은 장·재해를 유발할 수 있습니다. 따라서, 접지기술이 현장 기술자들의 관심사에만 머물지 않고 전기설비 관리자, 안전관리자 및 설계관계자에 이르기까지 그 중요성을 충분히 인식하여야 할 것입니다.

이에 본 연구원에서는 전기안전을 이룩하기 위한 유효한 수단인 접지문제에 있어서 일차적으로 중요하다고 판단되는 가장 기본적인 접지설계를 효과적으로 할 수 있는 설계프로그램을 작성하였습니다. 이는 지금까지 수작업에 의한 비효율적인 설계로 인한 시간적인 손실을 줄이고 보다 정확한 계산을 바탕으로 하는 설계를 하도록 하는 데 그 목적이 있습니다. 이 보고서를 현장의 설계 실무자들이 접지 시스템을 설계·시공하는 데 유용하게 활용할 수 있게 되기를 바라마지 않습니다.

1994. 12. 31

한국산업안전공단
산업안전연구원장

목 차

1. 서 론	1
1.1 연구배경	1
1.2 연구목표와 범위	2
1.2.1 연구목표	2
1.2.2 연구범위와 내용	2
2. 접지설계 기준	4
2.1 접지설계 고려사항	4
2.1.1 전위 상승 (GPR)	4
2.1.2 접촉전압과 보폭전압	5
2.1.3 전이전압과 접지계통 관계	6
2.2 IEEE 기준에 의한 접지설계	8
2.2.1 접지설계에 필요한 파라미터	8
2.2.2 설계 절차	9
3. 대지저항률 측정과 해석	22
3.1 대지저항률 측정 방법	22
3.1.1 Wenner 법	22
3.1.2 Schlumberger-Palmer 법	24

3.2 대지저항률 해석	25
3.2.1 대지저항률 고찰	25
3.2.2 2층 구조의 대지저항률 정식화	26
3.2.3 해석 알고리즘	27
4. 전위분포 해석	31
4.1 접지전극의 분할과 전류분포 계산	31
4.2 2층 구조 대지의 전위분포 해석	33
5. 결론 및 금후 과제	35
참고문헌	37
부록 : 접지설계 프로그램	39

1. 서 론

1.1 연구배경

접지시스템 설계의 목적은 전력계통 고장 상태시 고장전류로 인한 전위상승으로 발생할 수 있는 전기적 사고를 방지하는 데 있다. 따라서, 접지시스템은 고장시 전위상승 허용치가 안전기준의 허용범위 이내가 되도록 설계되어야 한다. 이러한 안전기준을 충족시키는 접지시스템의 특성은 접지대상 부지의 대지저항률 및 접지전극의 저항, 이와 연결된 전력계통의 구조 등에 따라 결정된다.

지금까지 접지시스템의 설계와 해석을 위한 가이드로서 ANSI/IEEE Standard 80이 미국을 비롯한 여러 나라에서 널리 이용되어 오고 있다. 우리 나라에서도 이 지침에 의한 접지설계가 보편적이다. 이 기준은 송전선로의 변전소를 대상으로 하여 많은 가정을 전제로 기술하고 있기 때문에 현장에서 직면하고 있는 여러 가지 복잡하고 특수한 고려사항을 정확하게 반영하지 못하는 약점을 가지고 있다. 따라서, 설계가 과장되고 시공 비용이 많이 드는 반면 안전하지 못한 접지설계가 되는 결과를 초래하기가 쉽다는 지적이 있다.

우리 나라의 경우, 대부분의 엔지니어링 회사나 전력회사 등에서 접지기술에 대한 정확한 해석에 대한 기술이 부족하다는 것이다. 대부분이 IEEE기준에서 제시하고 있는 식에 수치를 단순 반복적으로 대입하여 필요한 설계 데이터를 산출하고 있을 뿐이며, 여러 가지 복잡한 접지계통상의 문제에 대한 이론적 해석없이 경험적인 지식에 많이 의존하고 있는 실정이다. 그러나, 엔지니어들이 현재의 실무적인 설계문제에 대하여 피상적이고 단순한 관찰을 통해 내리는 결론과는 달리 접지 해석은 간단하지 않을 뿐만 아니라, 안전 측면에서 대단히 중요한 역할

을 하고 있다. 특히, 정보화 사회로의 이행이 급격히 이루어지고 있는 오늘의 시점에서 도시지역의 변전소나 전기설비는 이것이 시설되는 용지 확보의 어려움뿐만 아니라, 부근의 이해 당사자들이 전자파의 영향 등 환경과 안전성에 대한 명확한 평가 등을 요구하고 있는 추세임을 감안할 때 앞으로는 접지에 대해 심각하게 기술적인 검토를 하지 않으면 안되는 시점이라고 할 수 있다.

현재 우리 나라에서 직면하고 있는 접지설계나 시공상의 문제를 보다 더 깊이 살펴보면, IEEE 기준이나 기타 외국의 설계기준에 의해 설계를 하고 있는 회사는 극히 일부에 지나지 않는다. 자영업 수준의 전기설비 분야 설계종사자들은 접지설계에 대한 개념조차 없는 경우가 많다. 실무분야에서 접지설계를 실시한 자료가 아예 없는 경우가 허다한 실정임에 비추어 이에 대한 인식의 전환이 무엇보다 선행되어야 할 일이 아닐 수 없다.

본 연구에서는 이러한 우리 나라의 현실을 감안하여 접지의 중요성에 대한 인식을 제고시키고, 현장에서의 설계 관계자들이 접지설계 기술을 설계에 반영할 수 있도록 하며, 수작업에 의존하고 있는 설계과정을 프로그램화하여 보다 정확한 설계를 할 수 있도록 함으로서 전기안전이 근원적으로 확립될 수 있는 기틀을 마련하는 것이다.

1.2 연구목표와 범위

따라서, 본 연구는 이러한 우리 나라의 실정을 감안하여 다음과 같은 점에 연구목표를 두었으며, 연구의 범위와 내용은 아래와 같다.

1.2.1 연구목표

(1) IEEE Std. 80-1986에서 제시하고 있는 접지설계 과정을 프로그램으로 개

발한다.

(2) (1)의 설계과정에서 제시하지 않은 대지저항률 측정과 해석에 대한 수치 해석적 알고리즘 및 전위분포 해석에 대한 알고리즘을 개발한다.

1.2.2 연구범위와 내용

본 연구에서는 접지설계시 대부분 수작업에 의존하고 있는 우리 나라의 실정을 감안하여, 실무에 종사하고 있는 설계 종사자들이 IEEE에서 제시하고 있는 가이드에 의한 접지시스템 설계를 할 때 편리하게 이용할 수 있는 접지설계 프로그램을 개발하였다.

본 연구에서는 이에 더하여 첫째, 가이드에서 제시하지 않은 대지저항률 측정과 측정데이터로부터 대지의 구조를 정량적으로 해석할 수 있는 알고리즘을 개발하였다. 이 알고리즘을 통하여 지금까지 등한시하여 온 대지저항률의 정보 즉, 2층 수평구조의 대지에서 대지저항률이 달라지는 지층의 두께와 대지저항률 및 반사계수를 정확히 계산하여 접지설계에 반영할 수 있도록 하였다. 접지설계의 가장 첫 번째 단계에서 이루어지는 이 과정은 접지의 목적을 달성할 수 있는지의 성패에 직접적인 영향을 주기 때문에 이를 정확히 추정하는 것은 대단히 중요한 과정이라고 할 수 있다.

둘째, 인체의 전기안전기준 충족 여부를 가름하는 최대 메시전압을 보다 정확히 계산할 수 있는 전위분포 해석 알고리즘을 개발하였다. 이것은 IEEE에서 제시하는 메시전압이나 보폭전압을 구하는 간이 계산식보다 정밀한 계산방법으로, 전력계통의 고장시 접지시스템에 유입되는 고장전류로 인한 메시도체 주변의 전위분포의 양상을 해석할 수 있다.

2. 접지설계 기준

2.1 접지설계 고려사항

접지설계시에 접지계의 특성을 결정하기 위하여 계통의 고장을 상정하였을 때 발생하는 GPR과 접촉전압, 보폭전압 및 메시전압(최대허용값)을 그림 2.1에 나타난 바와 같이 비교하여 보자.

2.1.1 전위 상승 (GPR)

전류가 접지계통을 통하여 대지로 주입될 때 토양의 특성, 주로 대지저항률의 영향을 받는 접지저항에 의해 전위상승이 발생한다. 이 때의 전위상승 (GPR: Ground Potential Rise) 이란 접지전극과 무한히 먼 대지(Remote earth: 기준전위로 0전위를 말한다.)간에 발생하는 최대전압을 말한다.

GPR은 접지전극을 통해 대지로 흐르는 전류의 크기에 비례한다. 고장 발생시 고장전류의 전부가 접지계통으로 주입된다고 가정하면 과잉설계가 되기 때문에 설계대상 계통의 특성을 잘 고려하여야 한다. 주어진 주입전류에 대해 GPR은 토양의 대지저항률에 직접적으로 비례하기 때문에 접지설계시 토양 구조를 조사하고 접지대상 부지의 대지저항률을 측정하는 것이 대단히 중요하다. 더불어 계절 변동에 따른 대지저항률의 변화를 고려하여 설계시 반영하여야 한다.

주어진 주입전류에 따라 GPR은 대략 그리드의 면적에 반비례하며, 그리드의 매설깊이와 형상에 따라서도 GPR에 어느 정도 영향을 미친다.

지금, 그림 2.1은 변전소에서 무한히 먼 대지의 한 점과 접지전극사이의 전위 상승 측정의 개념을 나타내고 있다.

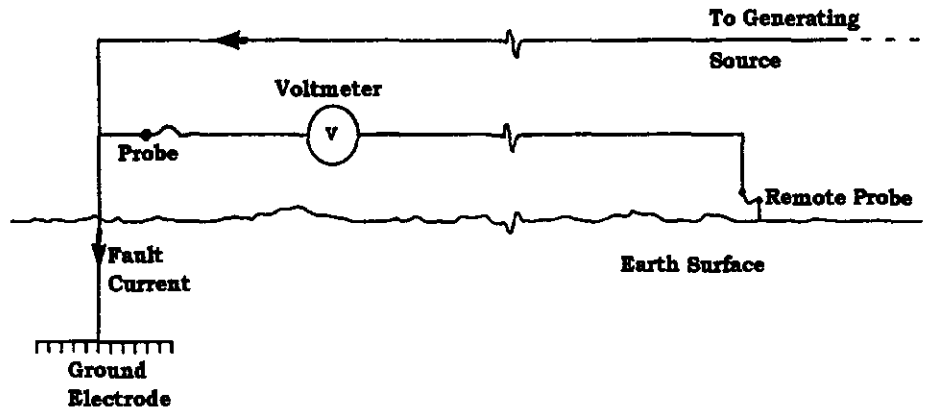


그림 2.1 접지전극의 GPR의 이론적 측정

GPR은 遠方点에 대하여 측정되므로 통신 케이블 등이 변전소 경계 내에서 원방점까지 매설되어 있는 경우에는 위험하게 된다. 여기서, "원방"이라는 말은 상대적인 것으로, 변전소의 규모에 따라 수 미터 떨어진 지점이 될 수도 있고 수 백 미터 떨어진 점일 수도 있으므로, 설계시에 이러한 점을 고려하여야 한다.

2.1.2 접촉전압과 보폭전압

접촉전압 (Touch Voltage)은 사람의 손이 접지 된 금속체에 접촉되어 있는 상태에서 서있는 지표면의 전위와 GPR간의 전위차를 말하며, IEEE 지침에서는 금속구조물과 대지간의 거리 1m의 전위차를 말한다.

보폭전압 (Step Voltage)은 사람이 서있는 상태에서 양 발을 1m 간격으로 벌렸을 때 지표면 위의 양발 사이에 걸리는 전압을 말한다. 접촉전압과 보폭전압에 대해 접지계통을 안전하게 하기 위해서는 2가지 접근이 가능하다. 첫째, 변전소 내와 그 경계의 어떤 지점의 접촉·보폭전압을 최소화하거나, 둘째 아스팔트나

자갈 같은 높은 저항률의 재료를 깔아 최대 허용 접촉·보폭전압을 높이는 것이다. 그런데 일반적으로는 이 2가지 방법이 동시에 사용된다. 이때 이들 재료를 정기검사하여야 하고, 접지계통의 경계밖에도 충분히 깔아야 한다.

접촉전압이란 변전소 접지계통과 지표면 위의 한 지점간의 전위차이므로, 지표면 전위가 가능한 변전소 GPR과 가까운 것이 이상적이다. 따라서, 접촉전압을 완화시키기 위해서는 전극의 도체 수를 증가시키면 가능하며, 이런 방법으로도 개선이 어려우면 접지면적을 증가시켜야 한다. 그러나, 도시지역의 변전소 부지는 경제적인 문제로 인하여 확대하기 어려우므로, 고저항률의 대지에서는 이를 효과적으로 완화시킬 수 있는 대책을 다각도로 강구하여야 한다.

보폭전압은 변전소 구역내인 경우, 접촉전압이 안전한 범위 내에 있는 경우라면 문제가 되지 않는다. 보폭전압이 접촉전압보다 작을 뿐 아니라, 통전 경로가 양 발사이이므로 보폭전압이 접촉전압보다 높더라도 허용이 된다. 일반적으로 보폭전압은 높은 전위경도가 나타나는 변전소의 경계주변만 관심의 대상이 된다. 전위경도의 첨두값은 접지계통의 규모에 반비례하므로, 보폭전압은 대규모 접지계통보다는 작은 규모의 것에 문제가 되기 쉽다.

2.1.3 전이전압과 접지계통 관계

轉移電壓(Transferred Voltage)은 어떤 특정 설비내에 서 있는 사람이 먼 거리에 접지 되어 있는 접지 된 도체에 접촉되거나, 또는 먼 위치의 사람이 접지계통에 연결된 도체에 접촉될 때 생기는 전압을 말한다.

그림 2.2는 철근 콘크리트 기초로 된 건물과 근처의 변전소 접지계통이 상호 분리되어 있을 때의 전위분포 관계를 나타내고 있다. 건물의 철근이 대지와 절연이 되어 있지 않는 한 동일한 전기적 특성을 가진 접지전극으로 모델링할 수 있다. 그림에서 2개의 전위 분포가 나타나 있는데, 변전소 그리드상의 등전위면은

GPR과 근접하여 있어 접촉전압은 문제가 되지 않으나, 빌딩의 기초에 나타난 전위와는 많은 차이가 있으므로, 고장시에는 이들 전위차에 의한 접촉전압이 크게 나타나므로 위험하게 된다.

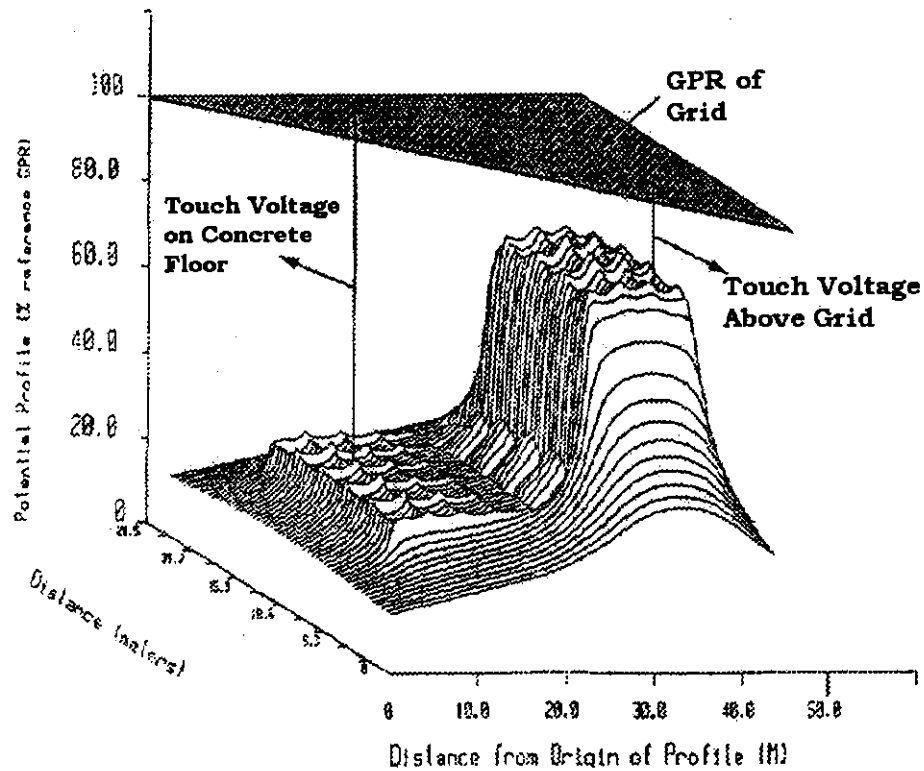


그림 2.2 변전소 접지전극과 인접한 건물간의 상호 영향

따라서, 안전측면을 고려할 때, 빌딩과 변전소 접지전극이 상호 연결되어 빌딩 기초가 변전소의 보조 접지극으로 사용되면 좋다. 또, 두 접지계통이 비교적 멀리 떨어져 있는 경우, 한 접지계통과 연결되어 있는 수도관 등이 다른 접지계

통 근처를 지나갈 때 접지가 되어 있지 않으면, 위험한 접촉전압이 나타나거나, 전이전압이 발생하게 되므로, 접지계통 구역으로 들어가는 모든 도체는 접지계통과 잘 접속되어 있는 것이 중요하다.

현재 우리 나라에서는 접지계통 근처에 매설된 수도관, 가스파이프 등의 도체에 대한 영향에 대하여 전혀 고려하지 않고 있으나, 앞으로는 상기한 이유 때문에 깊이 연구되어야 할 것이다.

2.2 IEEE 기준에 의한 접지설계

2.2.1 접지설계에 필요한 파라미터

접지를 시행하는 현장의 여건에 따라 달라지는 파라미터는 다음과 같은 데, 이들은 그리드 설계시 실질적으로 영향을 미치는 값이다.

(1) 최대그리드전류(I_g)

최대그리드전류는 접지전극의 저항, 고장전류의 분류계수와 앞으로의 계통의 확장 등을 고려하여 결정한다.

(2) 고장시간(t_f)과 인체전격시간(t_s)

특별한 경우가 아니면 이 두 파라미터는 같은 것으로 간주한다. 고장지속시간은 송전변전소나 배전 및 플랜트 변전소의 차단시간을 고려하여 결정하며, 보통 0.25~1 s 범위이다.

(3) 대지저항률

IEEE 80 기준에는 대지저항률의 측정이나, 해석에 대하여 특별한 지침이 없으므로, 본 연구에서는 별도로 이에 대하여 3장에서 자세히 다루었다.

(4) 자갈의 저항률

인체의 감전방지를 위하여 동절기의 지표면 동결로 인한 접지전극 주변의 대지저항률이 높아지는 것을 방지하기 위하여 변전소 구내의 지표면에 자갈이나 아스팔트 등을 깔아 준다. 이 때 자갈의 저항률은 $1,000 \sim 5,000\Omega$ 이내로 상정하며, $3,000\Omega$ 정도로 계산하는 것이 일반적이다.

(5) 전극 형상

접지전극은 접지목적과 시공장소에 따라 메시, 봉 등 여러 가지 형태의 것이 사용되고 있으나, 메시와 봉전극이 일반적이다. 메시인 경우, 도체간격은 $3\sim 15m$, 매설깊이는 $0.5\sim 2.5m$ 범위이며 전극이 크면, 접지저항, GPR, 메시전압 등을 경감시킬 수 있으므로, 전극의 설계가 대단히 중요하다.

2.2.2 설계 절차

(1) 설계 절차

스텝 1 변전소 접지설계를 위한 기초자료를 준비한다.

토양의 대지저항률 측정과 해석을 통하여 대지구조를 모델링하고 대지파라미터를 구한다. 본 연구에서는 대지저항률 해석에서 2층 구조의 대지를 등가 모델화한 알고리즘을 개발하여 파라미터(대지저항률, 지층두께, 반사계수)를 구할 수 있도록 하였다.

스텝 2 도체(접지선) 굵기를 구한다.

접지선에 전류가 흐르면, 주울 열이 발생하고 절연피복이 타거나 최악의 경우

용단될 수도 있다. 따라서, 충분한 전류용량, 즉 어느 정도의 도체 단면적이 필요하다. 도체 굵기는 전류, 통전시간, 온도, 재료의 특성값 등을 이용하여 식 (2.1)과 같은 IEEE 표준식을 이용하여 구한다.

$$A = I \sqrt{\frac{\frac{t_c \alpha_r \rho_r \times 10^4}{TCAP}}{\ln \left\{ 1 + \left(\frac{T_m - T_a}{K_0 + T_a} \right) \right\}}} \quad (2.1)$$

단, I: 접지선에 흐르는 전류[kA]

A: 도체 단면적[mm²]

T_m: 최대 허용온도[°C]

T_a: 주위온도[°C]

T_r: 기준온도[°C]

α₀: 0 [°C]일 때 도체의 열저항률

α_r: T_r 일 때 도체의 열저항률

ρ_r: T_r 일 때 도체의 저항률[μΩ/cm]

K₀: 1/α₀

t_c: 통전시간[s]

TCAP: 열용량 계수[J/cm³·°C]

이에 필요한 재료의 물리정수는 IEEE Std. 80-1986 의 표에서 구할 수 있다.

스텝 3 인체의 전기안전기준(접촉전압과 보폭전압)을 계산한다.

일반적으로 사람이 충전된 금속체 근처에서 한 손이 접촉되면, 사람이 서 있는 지표면과 금속체 사이에는 전위차가 생긴다. 따라서, 인체에는 전위차에 의한 전류가 흐르게 된다. 이 전류의 크기는 인체저항과 지표면 대지저항에 따라 달라진다.

(1) 인체 안전한계전류

인체의 안전한계전류를 I_E 라 하면, 최대허용 접촉전압과 보폭전압은 다음과 같이 나타낼 수 있다.

$$V_t = I_B(R_B + R_{FT}) \quad (2.2)$$

$$V_s = I_B + (R_B + R_{FS}) \quad (2.3)$$

단, V_t : 최대허용 접촉전압

V_s : 최대허용 보폭전압

R_B : 인체저항

R_{FS} : 양발사이의 저항(양발이 대지에 직렬로 되어 있는 상태의 저항)

R_{FT} : 양발사이의 저항(양발이 대지에 병렬로 되어 있는 상태의 저항)

인체의 안전한계 허용전류는 통전시간의 함수로 나타내며, IEEE 지침은 Dalziel의 식을 권장하고 있다. 지락사고시에 접지계통에 흐르는 전류를 I_A 라 하면, 안전한계전류 I_B 는 항상 $I_A < I_A$ 의 관계이어야 한다.

$$I_B = \frac{0.116}{\sqrt{t_s}} \quad (2.4)$$

여기서, t_s : 통전시간(초) (고장지속시간)

따라서, 고장시 고속도 차단이 안전측면에서 유리하며, 이 식은 0.03 ~ 3초 사

이에서만 적용할 수 있으나 이 범위 밖이면, 9mA로 한다.

0.116 : 체중 50kg 인 사람에 관계되는 상수로, 실제 접지설계시 이 값을 사용한다.

지락사고시에 접지계통에 흐르는 전류를 I_A 라 하면, 안전한계전류 I_B 는 항상 $I_A < I_B$ 의 관계이어야 한다.

그러나, IEC의 경우, IEEE와는 다른 곡선을 이용하고 있는데, 주요 차이점은 100명의 사람 중 99.5명이 전격을 당하지 않을 확률이고(IEEE), IEC의 경우, 99.86명이 전격을 당하지 않을 확률을 적용하고 있다는 점이다.

(2) 인체저항

IEEE 80에서는 지금까지 인체저항은 손과 양발 사이, 손과 손, 발과 발 사이를 전류 통과경로로 하여 1,000Ω으로 기준으로 정하고 있으나, 앞으로는 인체전류나 전압을 고려한 함수로 대체할 예정으로 있다. IEC Report 479-1은 인체의 저항을 접촉전압(건조상태)에 따라 다르게 적용하여야 한다는 주장을 하고 있으며, IEEE에서도 이에 대하여 동의하고 있는 실정이다.

최대허용 접촉전압과 보폭전압을 구하는데 필요한 발 저항은 지표면 대지의 토양특성에 따라 달라진다. 고려하고자 하는 접지계통의 토양이 균질 대지이고 지표면이 아스팔트나 자갈이 깔려 있지 않은 경우는 다음의 식을 이용하여 발 저항을 구한다.

$$R_{FT} = 0.5(3\rho) = 1.5\rho \quad (2.5)$$

$$R_{FS} = 2.0(3\rho) = 6.0\rho \quad (2.6)$$

그러나, 토양이 균질대지가 아니고(일반적으로 2층 구조) 지표면이 자갈 등이 깔려 있는 경우는 다음과 같은 식이 이용된다.

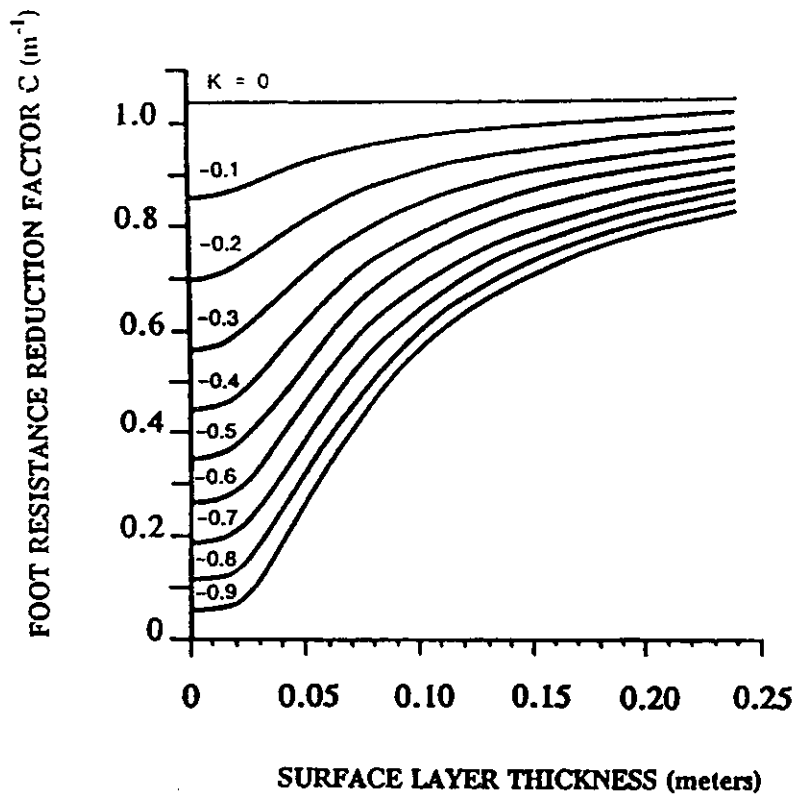


그림 2.2 자갈 층의 두께와 C 의 관계

$$R_{FT} = 1.5C\rho_s \quad (2.7)$$

$$R_{FS} = 6.0 C\rho_s \quad (2.8)$$

단, ρ_s : 상층 대지저항률

C : 발 저항 감소계수(그림 2.2에 의하여 구한다.)

그리고, 그림 2.2에서 토양 반사계수 K 는 다음 식에서 구한다.

$$K = \frac{\rho - \rho_s}{\rho + \rho_s} \quad (2.9)$$

(3) 인체에 인가되는 안전전압

변전소 구내에 메시전극이 매설되어 있을 때, 구내에 서 있는 작업자에 인가될 수 있는 전압은 보폭전압, 접촉전압, 메시전압이다.

인체저항은 앞서 설명한 바와 같이 1,000 Ω 으로 상정하고 있으며, 대지와 접촉된 발의 저항은 평균적으로 3 Ω 으로 가정하므로, 이를 토대로 접촉전압 및 보폭전압을 계산할 수 있다.

지금, 접촉전압이 손과 발 사이에 인가되고, 보폭전압이 양발사이에 걸릴 때 흐르는 전류를 각각 I_{BT} I_{BS} 라 하면, 다음과 같이 나타낼 수 있다.

$$I_{BT} = \frac{V_t}{1000 + 1.5\rho_s} \quad (2.10)$$

$$I_{BS} = \frac{V_s}{1000 + 6\rho_s} \quad (2.11)$$

따라서, 접촉전압과 보폭전압은 다음 식으로 구할 수 있다.

$$V_t \leq \frac{116 + 0.714 \rho_s}{\sqrt{t_s}} = E_t \quad (2.12)$$

$$V_s \leq \frac{116 + 0.696 \rho_s}{\sqrt{t_s}} = E_s \quad (2.13)$$

단, ρ_s : 지표면 대지저항률

(4) 자갈 층 대지저항률

위 식에서 접촉전압과 보폭전압은 자갈 층 대지저항률 ρ_s 에 따라 변화하므

로, 가급적 지표면을 고저항률로 유지하기 위하여 자갈 등으로 덮으며, 자갈 층의 저항률은 보통 3,000Ω 의 값을 취한다.

스텝 4 예비설계를 실시한다.

메시간격, 한 방향에서 본 메시도체 수, 메시의 총 길이, 메시 매설깊이 등을 대입하여 아래 스텝에 따라 예비설계를 하여, 안전기준을 초과하는 경우, 여러 가지 요소를 고려하여 이 기준을 만족할 때까지 반복계산을 수행한다.

스텝 5 그리드저항(접지저항)을 계산한다.

이상적으로는 접지저항이 0 Ω 이 가장 바람직하나, 실체는 있을 수 없는 문제이다. 일반적으로 대규모 송변전소인 경우 접지저항은 1Ω 이하가 요구되고 있으며, 플랜트나 기타 소규모인 곳에서는 1~5 Ω 이하로 설계되도록 하고 있다. 따라서, 접지설계라 하면, 이 접지저항 이하가 되도록 하는 설계를 말한다. 접지저항을 구하는 식은 여러 가지가 있으나 정확한 계산은 대단히 어렵기 때문에 개략적인 계산식으로 구하는 것이 일반적이다. 변전소 내에서 지표면 아래 0.25 ~ 2.5 m 정도의 깊이에 매설되는 메시전극에 대한 그리드저항의 계산에는 식 (2.14)와 같은 Sverak의 식이 이용된다.

봉전극을 같이 사용하는 경우에는 Schwarz식이 이용되고 있으나, 어느 식이든 대략적인 계산에서는 결과에서 별 차이가 없다. 이 식은 또한 예비설계에서 GPR을 추정하여 안전기준을 충족시키기 위한 매설도체 길이를 구하는데 필요한 식이다.

$$R_g = \rho \left[\frac{1}{L} + \frac{1}{\sqrt{20A}} \left(1 + \frac{1}{1 + h\sqrt{20/A}} \right) \right] \quad (2.14)$$

단, L : 매설도체의 길이[m]

ρ : 평균대지저항률[$\Omega \cdot m$]

A : 매시면적[m²]

h : 매설깊이[m]

스텝 6 최대 지락고장전류를 계산한다.

접지 사고의 형태는 여러 가지가 있다. 예를 들어, 1선 지락사고 (Single-line-to-ground)가 변전소 구내의 접지계통에서 일어나는 경우도 있고, 다른 접지계통에서 지락 되어 그 지락전류가 구내로 유입되는 경우도 있다. 또, 선로정수에 따라 지락전류의 값이 달라지기도 하므로, 지락전류의 값을 결정할 때는 이에 대한 계통의 종합적인 계산이 필요하다.

그러나, 여기서는 이에 대한 지락전류의 개략적인 계산법에 대해 설명하기로 한다. 지금, 지락전류 I_f 는 접지극에 흐르는 그리드전류를 I_g 라 하면, 다음과 같은 관계가 성립한다.

$$I_g = S_f \cdot I_f \quad (2.15)$$

여기서, S_f 는 분류계수(Current division factor)를 나타내며, 가공지선의 임피던스와 그리드저항에 반비례하여 흐르는데, 최적설계를 위하여서는 이에 대한 정확한 계산이 필요하나, 일반적으로 지락전류의 40~60 %가 접지망에 유입되는 것으로 추정하고 있다.

지락사고가 발생하면 보호계전기가 동작되어 회로를 차단한다. 이때, 전류의 과도적 상태가 발생하며, 특히 차단시간이 짧은 경우를 고려하고, 장치 계통이 확장될 것을 고려하면 최대지락전류(Maximum grid current)는 다음 식으로 나타낸다.

$$I_G = C_p D_f I_g \quad (2.16)$$

여기서, $C_p = 1.0 \sim 1.5$: 계통확장계수(Corrective projection factor)

D_f : 감쇠계수(Decrement factor), 0.5

1선 지락사고의 경우 영상전류 I_0 는 다음과 같은 간략 식으로 계산되며, 실용상 지장이 없다.

$$I_0 = \frac{E}{X_1 + X_2 + X_0} \quad (2.17)$$

단, E : 상전압

X_1 : 정상 리액턴스 [Ω /상]

X_2 : 역상 리액턴스 [Ω /상]

X_0 : 영상 리액턴스 [Ω /상]

일반적으로, $X_1 = X_2$

스텝 7 최대 메시전압을 구한다.

메시전극은 대규모 접지계통에서 많이 이용되는 접지전극이다. 메시를 포설하는 경우, 메시전극의 중심부와 4개 모서리(코너) 사이에 전위차가 발생한다. 메시전압은 이러한 경우를 고려한 전압으로, GPR과의 전위차를 말하며, 다음 식으로 구한다.

$$E_m = \rho K_m K_i I_G / L \quad (2.18)$$

단, K_m : 메시전극의 간격, 도체길이, 매설깊이, 도체 수에 의해 결정되는 메시간격 계수

K_i : 메시 보정계수(메시에 흐르는 전류의 크기가 각각 다르며, 이를 고려한 값)

I_G : 최대 지락전류[A]

L : 매설도체의 길이[m]

여기서, 메시간격계수 K_m 은 식(2.19)에 의하여 계산된다.

$$K_m = \frac{1}{2\pi} \left[\ln \left(\frac{D^2}{16hd} + \frac{(D+2h)^2}{8Dd} - \frac{h}{4d} + \frac{K_{ii}}{K_h} \ln \frac{8}{\pi(2n-1)} \right) \right] \quad (2.19)$$

단, K_{ii} : 메시에 봉전극을 병용한 경우는 이 값은 1 이며, 메시만의 경우는 다음 식에서 구할 수 있다.

$$K_{ii} = \frac{1}{(2n)^{2/n}}$$

$$K_h = \sqrt{1 + \frac{h}{h_0}}$$

h_0 : 1[m]

h : 메시전극의 매설깊이[m]

D : 메시간격[m]

n : 한 방향에서 본 도체수[m]

d : 메시도체 지름[m]

메시 보정계수 K_{ii} 는 메시도체 수에 의해 결정되며, K_i 는 다음과 같이 구한다.

$$K_i = 0.656 + 0.172n \quad (2.20)$$

메시에 봉전극을 추가하여 병용하는 경우에는 식 (2.18)은 다음과 같이 나타낼 수 있다.

$$E_m = \frac{\rho I_G K_m K_i}{L_c + 1.15 L_r} \quad (2.21)$$

단, L_c : 메시도체 길이[m]

L_r : 봉 길이[m]

스텝 8 최대 보폭전압을 계산한다.

메시가 대규모인 경우 전류는 메시의 중심부보다 모서리 부분으로 많은 전류가 흐르게 되고, 이에 따라 중심부와 모서리 사이에는 전위차가 발생하며, 전위경도가 크게 된다. 메시의 모서리에서 1[m] 떨어진 곳의 전압을 보폭전압으로 정의하여 다음 식으로 나타낸다.

$$E_s = \rho K_s K_i I_G / L \quad (2.22)$$

단, K_s : 메시간격계수로 다음 식으로 구한다.

$$K_s = \frac{1}{\pi} \left[\frac{1}{2h} + \frac{1}{D+h} + \frac{1}{D} (1 - 0.5^{n-2}) \right] \quad (2.23)$$

메시전극에 봉전극을 병용하는 경우는 다음 식으로 나타낸다.

$$E_m = \frac{\rho I_G K_s K_i}{L_c + 1.15 L_r} \quad (2.24)$$

(2) 설계 흐름도

이상의 설계절차를 흐름도로 그리면 그림 2.3과 같다. 본 연구에서의 설계 프로그램은 이 흐름도에 따르도록 개발되었다. 여기서, 첫 번째 스텝인 대지저항률

은 3장에서 그 알고리즘을 다루었으며, 이 알고리즘을 이용하여 상층의 대지저항
률과 변곡점이 발생하는 깊이 및 반사율을 정량적으로 해석할 수 있다.

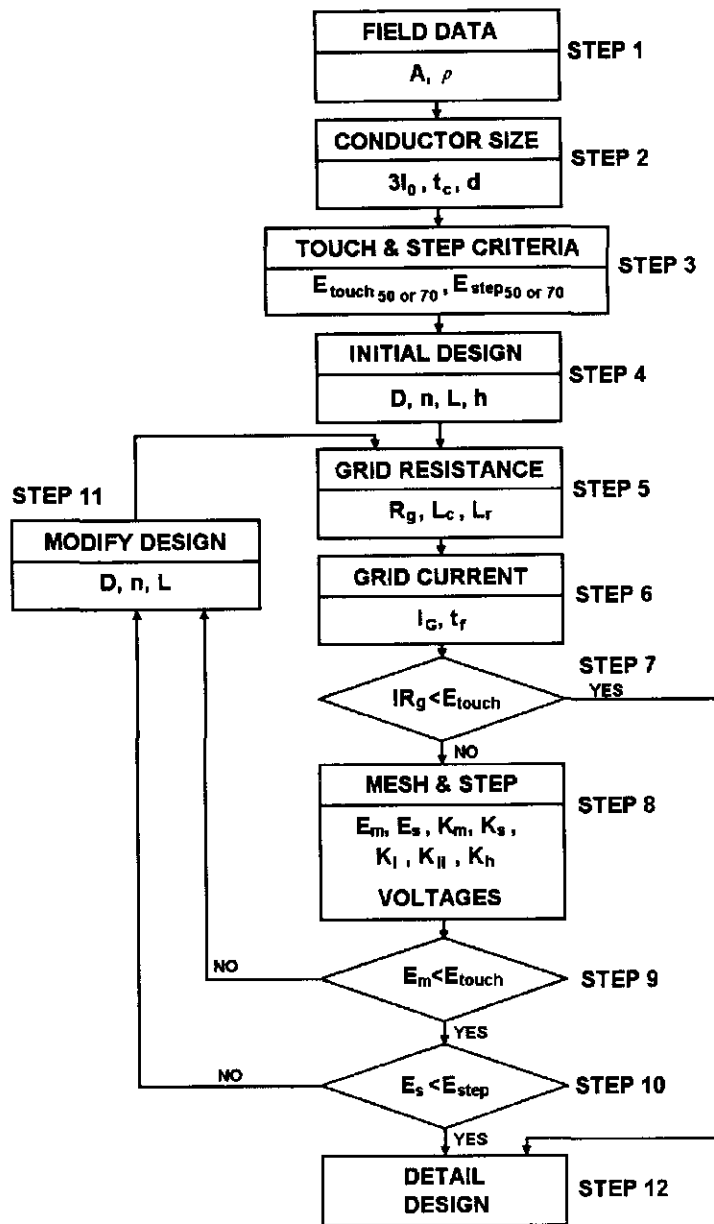


그림 2.3 접지설계 흐름도

3. 대지저항률 측정과 해석

지락 등으로 인한 고장전류가 접지계통에 흐를 때 이의 특성을 정확히 예측하기 위해서는 대지저항률(Soil Resistivity)을 정확히 구하는 것이 중요하다. 접지설계의 가장 첫 번째 단계에서 이루어지는 이 과정은 접지의 목적을 달성할 수 있는지의 성패에 직접적인 영향을 주는 파라미터(대지저항률과 지층의 두께)를 구하는 것이다. 대지저항률 측정은 지표면의 특정 지점에 전류를 가하여 이때의 전기적 응답으로부터 도전 매개체(대지)의 전기적 특성을 결정하는 것으로, Wenner법이 널리 이용되고 있다. 그러나, 대지는 일반적으로 지층에 따라 대지저항률이 다른 다층구조이며, 지표면의 대지저항률은 계절과 온도, 습도 등의 영향을 받는 종속변수이므로 이를 정확하게 추정하는 것은 대단히 복잡한 문제이다. 일반적으로 대지의 하층 구조에 대한 복잡한 해의 존재는 접지전극의 응답을 결정하는데 그다지 중요하지 않으므로, 대부분의 경우 실제의 대지구조는 등가 2층 구조로 모델화 할 수 있다.

3.1 대지저항률 측정 방법

3.1.1 Wenner 법

대지저항률을 측정하는데 가장 일반적인 방법은 Wenner법이다. 이 방법은 측정기가 적합하기만 하면, 정확한 대지모델에 필요한 데이터를 측정할 수 있다. 이 측정법은 측정이 간단하고 간편하며, 이에 적합한 휴대용 측정기가 여러 가지 개발되어 있으므로, 효과적으로 측정할 수 있는 표준측정법으로 권장되고 있다.

그림 3.1은 Wenner의 4전극 배열을 나타내고 있다. 4개의 전극이 일직선으로 등간격 배열되어 있으며, 전극의 매설깊이는 일반적으로 2개의 전극간격의 10% 이하이다.

지금 전극 C_1, C_2 사이에 전원을 접속하여 대지에 전류를 흘려서, 전극 P_1, P_2 간에 발생하는 전위차를 측정한다.

전위차의 측정값을 흘려준 전류로 나누면, 접지저항을 구할 수 있다. 이때 단위 길이당 대지저항률을 ρ 라 하면, 다음 식으로 나타낼 수 있다.

$$\rho = \frac{4\pi aR}{1 + \frac{2a}{\sqrt{a^2 + 4b^2}} - \frac{a}{\sqrt{a^2 + b^2}}} \quad (3.1)$$

단, a : 전극간격

b : 전극 매설깊이

여기서, 전극 매설깊이는 $0.1a$ 이므로, $b=0$ 으로 놓아도 실용상 지장이 없다. 그러므로, 위 식은 다음과 같이 표현할 수 있다.

$$\rho = 2\pi aR \quad (3.2)$$

이 식에서 깊이 a 의 변화에 따른 토양의 평균대지저항률을 구할 수 있다.

전극간격을 변화시키면서 대지저항률을 측정하여 간격과 대지저항률의 관계를 그래프로 그리면, 대지저항률이 뚜렷하게 다른 지층이 나타나는 것을 알 수 있다.

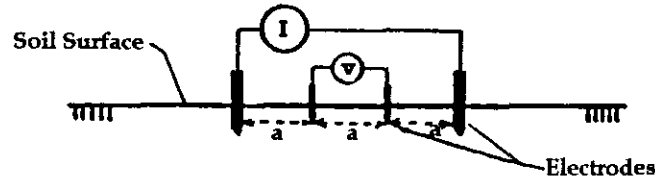


그림 3.1 Wenner 법에 의한 대지저항률 측정

3.1.2 Schlumberger-Palmer 법

Wenner법의 결점은 측정간격이 비교적 클 경우 전위 값이 급격히 작아진다는 것이다. 지금까지의 접지저항 측정기는 이러한 저전위 값을 측정하는데 부적합한 것으로 알려져 있다. 따라서, 전류전극간의 간격이 큰 대지저항률을 측정할 수 있으려면 그림 3-2와 같은 배열이 적합하다. 전압전극을 대응되는 전류전극에 더 가까이 배치함으로써 측정되는 전위 값을 증가시킬 수 있다. 전극 매설깊이 b 가 측정간격 d , c 에 비하여 작은 경우의 계산식은 다음 식과 같다.

$$\rho = \frac{\pi c(c+d)R}{d} \quad (3.3)$$

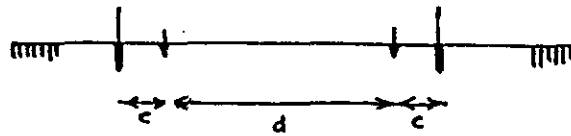


그림 3-2 Schlumberger-Palmer 법에 의한 측정

3.2 대지저항률 해석

3.2.1 대지저항률 고찰

현장에서 측정한 대지저항률 값에 대한 해석은 상당히 어려운 일에 속한다. 실제의 대지구조는 지층별로 서로 다르기 때문에 대지저항률의 변화가 복잡한 경우가 많다. 대지저항률 해석방법에는 다음과 같은 2가지가 있다.

(1) Wenner 배열에 따라 깊이를 변화시키면서 측정된 대지저항률과 전극간격을 이용하여 그린 그래프에 의하여 대지구조를 추정하는 방법

(2) 측정값으로부터 등가 대지구조로 계산하는 해석적 방법

전자의 방법은 컴퓨터가 등장하기 전에 널리 이용되었던 것으로, 주로 경험적 해석방법이라 할 수 있다. 이 방법에는 대지저항률 그래프에서 대지저항률이 다른 지층에 도달할 때 생기는 변곡점을 이용하는 Gish & Rooney법과, 저항률이 다른 하층의 깊이는 변곡점이 발생한 전극간격의 2/3을 취한 값으로 해석하는 Lanczster-Jones법이 있다. 그러나, 그래프를 가지고 동일한 대지저항률의 두께를 구하는 것이 결코 용이한 일이 아니다.

후자는 대지를 2층 등가대지로 가정하여 이론적 대지저항률과 대지파라미터(대지저항률, 지층두께, 반사계수)간의 수학적 관계를 구하여 측정값과 비교추정하는 방법으로, 정량적으로 해석할 수 있는 가장 적합한 방법이다.

IEEE Std. 80에서는 대지를 균질(Homogeneous Soil)로 취급하여 대지저항률을 초기 값으로 대입하기 때문에 대지저항률의 측정과 해석에 대하여 구체적으로 기술한 가이드라인이 없는 실정이다. 따라서, 본 연구에서는 현장 측정에서

얻은 대지저항률 값을 이용한 해석적 방법의 알고리즘을 개발하여 제시하였다.

3.2.2 2층 구조의 대지저항률 정식화

접지대상 부지를 그림 3.3과 같이 등가 2층 구조 대지로 모델링할 때 대지저항률은 다음 식으로 주어진다 2층 구조로 해석할 때 수평 구조, 수직 구조, Exponential 구조의 3가지로 등가 모델화할 수 있으나, 가장 일반적인 방법은 수평구조로 해석하는 것이다.

$$\rho_{aj} = \rho_1 \left[1 + 4 \sum_{n=1}^{\infty} K^n \left(\frac{1}{\sqrt{A_j}} - \frac{1}{\sqrt{B_j}} \right) \right] \quad (3.4)$$

여기서,

ρ_a : 겉보기 대지저항률

ρ_1 : 상층 대지저항률

$$A_j = 1 + \left(\frac{2nh}{a_j} \right)^2$$

$$B_j = A_j + 3$$

h : 상층 두께

a_j : 전극 간격

$$K = \frac{\rho_2 - \rho_1}{\rho_2 + \rho_1}$$

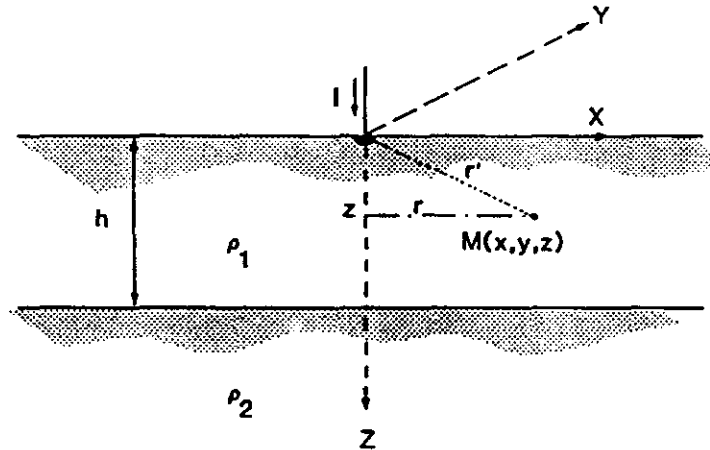


그림 3.3 2층 구조 대지 모델

3.2.3 해석 알고리즘

대부분의 컴퓨터에 의한 수치해석법은 기존의 방정식을 사용하는 것으로 계산속도, 알고리즘의 안정성, 수렴특성 등을 고려하여 선택한다. 본 연구에서 개발한 대지저항률 해석 프로그램은 Steepest-descent 법을 사용하였다. 그 이유는 다음과 같다.

① 이 방법은 연구결과 대단히 안정적이고 신뢰할 만한 방법임이 입증되었다. 미리 추정된 대지모델의 초기 값을 요구하는 것이 아니므로, 경험이 많지 않은 엔지니어도 사용할 수 있다.

② 수행하기가 쉽다.

Steepest-descent 법은 그림 3.4와 같이 함수 $\phi(x, y)$ 를 해석함으로서 나타낼 수 있다. 이 함수의 Gradient 는 초기점 M에서 계산된다. Gradient vector로 정

의된 방향을 따라 함수가 감소하도록 x와 y의 값을 선정한다. 이 과정은 초기 방향을 따라가는 함수가 증가하기 시작할 때까지 반복되다가 현재의 축(x, y) 이 함수의 최소 gradient 가될 때 멈추게 된다.

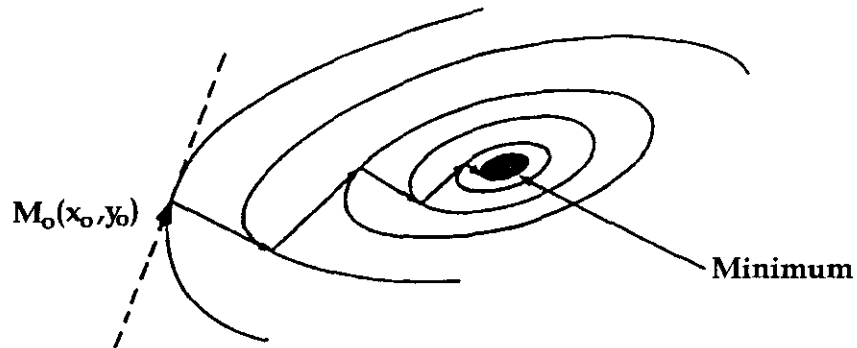


그림 3.4 Steepest-Descent 법

어떤 기준에 의해 계산된 값과 측정값간에 차이가 최소가 되는 가장 적합한 대지모델을 찾는 것이 필요하며, 이 기준으로 최소자승법(Least-square criterion)이 이용된다.

$$\phi(\rho_1, K, h) = \sum_{j=1}^n \left[\frac{\rho^0(a_j) - \rho(a_j)}{\rho^0(a_j)} \right] \quad (3.5)$$

여기서,

$\rho^0(a_j)$: 대지저항률 측정값

$\rho(a_j)$: 대지저항률 계산값

식(3.5)에서 ϕ 가 최소가 될 때 실제 값과 가장 잘 부합되는 값이며, 최소가 되게 하는 ρ_1, K, h 의 값은 Steepest-descent 알고리즘으로 결정된다.

여기서, Gradient vector V 는 다음과 같이 정의된다.

$$V = \left(\frac{\partial \phi}{\partial \rho_1}, \frac{\partial \phi}{\partial h}, \frac{\partial \phi}{\partial K} \right) \quad (3.6)$$

Potential vector 의 각 성분은 식 (3.5)에 의해 결정된다.

$$\begin{aligned} \frac{\partial \phi}{\partial \rho_1} &= -2 \sum_1^n \left(\frac{\rho^0}{\rho^{\circ 2}} \right) \frac{\partial \rho}{\partial \rho_1} \\ \frac{\partial \phi}{\partial K} &= -2 \sum_1^n \left(\frac{\rho^0}{\rho^{\circ 2}} \right) \frac{\partial \rho}{\partial K} \\ \frac{\partial \phi}{\partial h} &= -2 \sum_1^n \left(\frac{\rho^0}{\rho^{\circ 2}} \right) \frac{\partial \rho}{\partial h} \end{aligned} \quad (3.7)$$

$$\begin{aligned} \Delta \rho_1 &= -\tau \frac{\partial \phi}{\partial \rho_1} \\ \Delta K &= -\tau \frac{\partial \phi}{\partial K} \\ \Delta h &= -\tau \frac{\partial \phi}{\partial h} \end{aligned} \quad (3.8)$$

단, τ 는 최소화하기 위하여 필요한 값(V 의 p.u로 표시한다.)

$$\begin{aligned} \Delta \phi &= \frac{\partial \phi}{\partial \rho_1} \Delta \rho_1 + \frac{\partial \phi}{\partial K} \Delta K + \frac{\partial \phi}{\partial h} \Delta h \\ \text{또는, } \Delta &= - \left[\tau \left[\frac{\partial \phi}{\partial \rho_1} \right] + \tau \left[\frac{\partial \phi}{\partial K} \right] + \left[\frac{\partial \phi}{\partial h} \right] \right] \end{aligned} \quad (3.9)$$

여기서, 최소화를 만족하는 조건은 $\Delta \phi = 0$ 이거나, $\Delta |\phi| < \varepsilon$ 일 때이다.

위와 같은 과정을 정리하여 보면, Steepest descent 알고리즘의 주요 스텝은 다음과 같다.

1. 초기값 ρ_1, K, h 를 추정한다.
2. τ 를 계산한다.
3. $\Delta\rho_1, \Delta K, \Delta h$ 를 결정한다.
4. 다음 식으로 부터 미지수 벡터를 수정한다.

$$\rho_1(i) = \rho_1(i-1) + \Delta\rho_1$$

$$K(i) = K(i-1) + \Delta K$$

$$h(i) = h(i-1) + \Delta h$$

5. $|\Delta\psi|$ 를 계산하여 계산의 허용오차 ε 와 비교한다.

- a. $|\Delta\psi| < \varepsilon$ 이면 계산을 종료하고,
- b. $|\Delta\psi| > \varepsilon$ 이면 step 2 로 가서 반복 계속한다.

식 (3.9)에서 $\Delta\psi$ 를 계산하려면 ψ 의 편도함수를 알아야 한다. 이 값은 식 (3.7)에서 결정되며, 2층 대지 길보기 대지저항률 함수의 편도함수는 식 (3.4) 에서 구한다. 이러한 계산은 다음과 같다.

$$\begin{aligned} \frac{\partial \rho}{\partial \rho_1} &= 1 + 4 \sum_{n=1}^{\infty} k_n [(1 - n(1-K)^2/2K) (A^{-1/2} - B^{-1/2})] \\ \frac{\partial \rho}{\partial \rho_1} &= 4P_1 \sum_{n=1}^{\infty} nK^{n-1} (A^{-1/2} - B^{-1/2}) \\ \frac{\partial \rho}{\partial \rho_1} &= \frac{-16\rho_1 h}{a^2} \sum_{n=1}^{\infty} K^n [A^{-3/2} - B^{-3/2}] n^2 \end{aligned} \quad (3.10)$$

$$\text{단, } A = 1 + (2nh/a)^2$$

$$B = A + 3$$

4. 전위분포 해석

균질 또는 2층 대지에 고장전류가 주입될 때 접지전극에는 전위가 생기며, 이것은 무한히 멀리 떨어진 대지의 전위가 0이라 할 때, 대단히 높은 전위경도라 할 수 있다. 메시전극에 분포되는 전위가 거의 같다고 하더라도, 전극과 토양에 흐르는 전류밀도는 변화한다. 등간격의 그리드인 경우 전류가 전극의 중심부보다는 전극의 모서리로 많이 흐르는 등의 이유 때문이다.

그러나, 이러한 메시주변의 지표면 전위분포 계산에서는 다음과 같은 가정이 필요하다.

- (1) 접지시스템은 등전위면이다.
- (2) 토양은 2층 구조이다.

실제 접지시스템은 대부분 수평, 수직이거나 이와 비슷한 구조이므로 여기서는 수평으로 간주한다.

4.1 접지전극의 분할과 전류분포 계산

그림 4.1과 같이 전극이 사각형 형태일 때 보다 정확한 전위분포를 계산하기 위하여 전류밀도가 동일하다고 생각되는 길이로 작게 분할한다. 이때 길이를 작게 나누면 나눌수록 결과는 정확해진다. 그러나, 일반적으로 접지도체의 최대길이를 1/10로 분할할 때 정확도가 높다는 연구결과가 있다. 결국 주어진 정확성에 적합한 최소 세그먼트(Segment) 수는 도체를 나누는 방법에 달려 있다. 다시 말하자면 세그먼트를 최소로 하면서 정확한 결과를 얻을 수 있는 최적분할(Optimum subdivision)을 하는 것이 필요하다.

예를 들어, 그림 4.1에서 보는 바와 같은 분할한다고 하자. 지금 동일한 전류밀도를 갖는 세그먼트에 대하여 번호를 부여한다. 이와 같이 분할하여 번호를 부

여하면 컴퓨터 프로그램을 단순화할 수 있으며, 번호를 매기는 데 대한 특별한 규칙이 있는 것은 아니고 설계자가 임의로 부여할 수 있다.

분할 방법은 그리드의 모서리 부분은 조밀하게, 그리드의 중앙부는 간격을 넓게 하고 상하와 좌우를 동일한 간격으로 나눈다. 또, 그리드의 중심에서 상하가 대칭이 되게 하고, 좌우간이 대칭이 되도록 분할한다. 이렇게 하여 1~8 segment는 동일한 길이로 대칭관계이므로 동일한 전류밀도를 갖는다. 마찬가지로, 9~16, 17~20, 21~24 세그먼트 그룹은 각각 같은 전류밀도를 갖게 된다. 따라서, 그리드 도체가 24개로 분할되었더라도 전류밀도가 다른 것은 4개이므로, 이를 구하여 더하면 그리드 전류가 된다.

2		18		4	
7	10	15	22	18	8
19	9	23	21	24	20
5	1	13	17	14	6

그림 4.1 도체의 분할

분할된 각각의 전류를 구하기 위해서 다음과 같은 4개의 연립방정식을 세우면 다음과 같다.

$$\begin{aligned}
 r_{11}i_1 + r_{12}i_2 + r_{13}i_3 + r_{14}i_4 &= V_1 \\
 r_{21}i_1 + r_{22}i_2 + r_{23}i_3 + r_{24}i_4 &= V_2 \\
 r_{31}i_1 + r_{32}i_2 + r_{33}i_3 + r_{34}i_4 &= V_3 \\
 r_{41}i_1 + r_{42}i_2 + r_{43}i_3 + r_{44}i_4 &= V_4
 \end{aligned} \tag{4.1}$$

단, r_{jj} : 자기저항

r_{jk} : 상호저항

이를 매트릭스 형태의 일반식으로 나타내면, 다음과 같이 된다.

$$[V] = [R] [I] \tag{4.2}$$

$$\text{단, } V = \begin{bmatrix} V_1 \\ V_2 \\ \vdots \\ V_n \end{bmatrix} \quad I = \begin{bmatrix} I_1 \\ I_2 \\ \vdots \\ I_n \end{bmatrix}$$

V_j : Segment j의 전압

I_j : Segment j로부터 전류

$[R]$: $n \times n$ symmetric matrix

4.2 2층 구조 대지의 전위분포 해석

식 (4.2)에서 각 세그먼트 그룹별로 전류를 계산하면, 이 값으로부터 어떤 점에서의 전위를 구할 수 있다. 길이 L인 도체에 전류 I가 흐를 때의 전위는 다음 식에서 구하여 진다.

$$V = \frac{I\rho}{2\pi L} \ln \frac{\sqrt{x^2+y^2+D^2}+x}{\sqrt{(x-L)^2+y^2+D^2}+x-L} \quad (4.3)$$

단, ρ : 대지저항률(균질대지)

D: 전극 매설깊이

x, y : 임의의 점의 좌표

지금 대지가 2층 구조의 대지이고 상층의 두께가 h, 이때의 대지저항률이 ρ_1 , 하층의 대지저항률이 ρ_2 이라 하고, 전극이 상층에 있는 경우 식 (4.3)은 식 (4.4)와 같이되며, 하층에 있는 경우는 식(4.5)과 같이 나타낼 수 있다.

(1) 전극이 상층에 있을 때

$$V = \frac{I\rho}{2\pi L} \left[\ln \left(\frac{\sqrt{x^2+y^2+D^2}+x}{\sqrt{(x-L)^2+y^2+D^2}+x-L} \right) + \sum_{n=1}^{\infty} K^n \right. \\ \left[\ln \left(\frac{\sqrt{x^2+y^2+(2nh+D^2)}+x}{\sqrt{(x-L)^2+y^2+(2nh+D^2)}+x-L} \right) \right. \\ \left. + \ln \left(\frac{\sqrt{x^2+y^2+(2nh-D^2)}+x}{\sqrt{(x-L)^2+y^2+(2nh-D^2)}+x-L} \right) \right] \quad (4.4)$$

(2) 전극이 하층에 있을 때

$$V = I\rho_1(1+K) \left[\ln \left(\frac{\sqrt{x^2+y^2+D^2}+x}{\sqrt{(x-L)^2+y^2+D^2}+x-L} \right) + \sum_{n=1}^{\infty} K^n \right. \\ \left. \ln \left(\frac{\sqrt{x^2+y^2+(2nh-D^2)}+x}{\sqrt{(x-L)^2+y^2+(2nh-D^2)}+x-L} \right) \right] \quad (4.5)$$

5. 결론 및 금후 과제

본 연구의 수행결과는 다음과 같이 요약할 수 있다.

1. IEEE Guide 80에서 제시하고 있는 접지설계의 전 과정을 프로그램으로 개발하였다. 이는 지금까지 수작업에 의존하여 오던 우리 나라 접지설계 분야에 많은 도움이 될 수 있을 것이다.

2. 접지설계의 첫 단계에서 가장 중요한 대지저항률의 측정기법을 다루었으며, 측정값에서 대지구조의 특성을 해석할 수 있는 알고리즘을 제시였다. 이를 토대로 접지설계 종사자들이 보다 정확한 접지설계를 할 수 있을 것이다.

3. 전위분포 해석기법을 제시함으로서 종전까지 최대 메시전압만으로 안전성 여부를 판단하던 설계에서 나아가 설계하고자 하는 접지시스템의 전위분포를 포괄적으로 파악할 수 있게 되었으며, 이를 토대로 접지계통과 인접한 수도관, 가스관, 통신선로 등과의 상호 영향을 평가할 수 있는 기초가 될 수 있을 것으로 판단된다.

우리 나라에서의 접지에 대한 연구는 현재 겨우 시작의 단계에 있으나, 향후 계속적인 연구가 수행된다면, 접지설계에 대한 독자적인 결과의 도출이 가능할 것이다.

따라서, 앞으로 연구과제로 검토되어야 할 사항은 다음과 같이 요약할 수 있다.

- 본 연구에서 개발한 접지설계 프로그램을 기본적인 토대로 하여 계속 내용을 보완하여 현장에서 발생하는 다양한 기술적 요소를 반영할 수 있도록 하여야 한다.

- 접지면적이 협소한 도시지역의 접지전극으로 효용성이 있는 봉전극에 대한

접지효과의 정량적 규명을 위한 연구가 필요하다.

- 우리 나라의 계절변동에 따른 대지저항률에 대한 기초 연구의 필요성이 제기된다.

- 과도상태에서의 접지특성에 대한 규명이 요망된다.

참 고 문 헌

- [1] IEEE Std. 80, IEEE Guide for Safety in AC Substation Grounding, 1986
- [2] IEEE Std. 81, IEEE Guide for Measuring Earth Resistivity, Ground Impedance and Earth Surface Potentials of a Ground System, 1983
- [3] Sunde, E. D., Earth Conduction effects in Transmission Systems, Dover Pub. 1968
- [4] Tagg, G. F., Earth Resistance, George Newnes Limited, 1964
- [5] Meliopoulos, A. P. S., Power System Grounding and Transients, Marcel Dekker, Inc., 1988
- [6] EPRI, Transmission Line Grounding, Vol.1, Research Project 1494-1, Report EL 2695, Principal Author F. Dawalibi, 1982
- [7] EPRI, Analysis Techniques for Power Substation Grounding Systems, Vol. 1, Research Project 1494-2, Report EL 2682, 1982
- [8] 高橋健彦, 이형수역, 接地設計 入門, 동일출판사, 움사, 1993
- [9] 이형수, 이관형, 이봉용, 심건보, “최적접지설계를 위한 대지 파라미터의 추정”, '94 대한전기학회 추계학술대회 논문집, 1994, pp.61-63
- [10] Alamo, J. L. del, “A Comparison among Eight Different Techniques to Achieve an Optimum Estimation of Electrical Grounding Parameters in 2-Layer”, IEEE Trans. Power Delivery, Vol. 8, No. 4, 1993
- [11] Dawalibi, F. and Mukhedkar, D., “Optimum Design of Substation Grounding in Two-Layer Earth Structure; Part I - Analytical Study, Part II - Comparison Between Theoretical and Experimental Results, Part III - Study of Grounding Grids Performance and New Electrodes Configuration”, IEEE Trans. PAS, Vol.

PAS-94, No. 2, 1975, pp. 252-261, 262-266, 267-272

- [12] Dawalibi, F. and Mukhedkar, D., "Parametric Analysis of Grounding Systems", IEEE Trans. on PAS, Vol. PAS-98, No. 5, 1979, pp. 1659-1668
- [13] Robert J. Heppe, "Computation of Potential at Surface above an Energized Grid or Other Electrode, Allowing for Non-uniform Current Distribution", IEEE Trans. on PAS, Vol. PAS-98, No. 6, 1979
- [14] Garrett, D. L. and Holley H. J., "Calculation of Substation Ground Resistance Using Matrix Techniques", IEEE Trans. on PAS, Vol. PAS-99, No.5, 1980, pp. 2008-2011
- [15] SES Ltd., Power System Grounding and Electromagnetic Field, Technical Seminar, San Francisco, 1994

부록 : 접지설계 프로그램

```
#include <dos.h>
#include <dir.h>
#include <math.h>
#include <ctype.h>
#include <conio.h>
#include "gtypedef.h"
#include "pd_menu.h"
#include "plot3d.h"
#include "hanin.h"

#define NO_SELECT "NO_SELECT"
#define TAB_NO 10
#define MLEN 8
#define MAX_FILE_NO 20

#define X0 136
#define Y0 91
#define X1 514
#define Y1 389

const char *tab_item[][MAX_LEN] = {
    "Material conductivity [%]",
    "alpha_r factor @ 20 °C",
    "K (1/alpha_0) @ 0 °C",
    "Fusing temperature [°C]",
    "rho_r @ 20 °C [ $\mu\Omega$ /cm]",
    "TCAP factor effective value [ $J/cm^3/°C$ ]",
};

float get_data(int x, int y, char *form)
{
    char buffer[MLEN + 1];

    memset(buffer, ' ', MLEN);
    buffer[MLEN] = '\0';
    hgotoxy(x, y);
    hgetnumdata(buffer, form);
    return atof(buffer);
}
```

```

void draw_3d_plate(int x0, int y0, int x1, int y1)
{
    setfillstyle(SOLID_FILL, DARKGRAY);
    bar(x0, y0, x1 - 2, y1 - 2);
    rectangle(x0, y0, x1 - 2, y1 - 2);
    setcolor(LIGHTGRAY);
    setlinestyle(SOLID_LINE, 1, 1);
    moveto(x1 - 1, y0 + 1);
    lineto(x1 - 1, y1 - 1);
    lineto(x0 + 1, y1 - 1);
    moveto(x0 + 2, y1);
    lineto(x1, y1);
    lineto(x1, y0 + 2);
}

void no_job_error(void)
{
    void far *buf;
    int          xx0, yy0, xx1, yy1;
    int          x0 = 24, y0 = 13, x1 = 57, y1 = 17;

    xx0 = x2px(x0);
    yy0 = y2py(y0);
    xx1 = x2px(x1);
    yy1 = y2py(y1);
    buf = farmalloc(imagesize(xx0, yy0, xx1, yy1));
    getimage(xx0, yy0, xx1, yy1, buf);
    draw_3d_plate(xx0, yy0, xx1, yy1);
    sound(500);
    delay(200);
    nosound();
    hsetbkcolor(DARKGRAY);
    hsetcolor(LIGHTCYAN);
    hgotoxy(x0 + 2, y0 + 1);
    hputs("JOB이 선택되어있지 않습니다 !");
    hgotoxy(x0 + 2, y0 + 2);
    hputs("아무 키나 치세요...");
    hrestartcursor();
    hgetch();
    putimage(xx0, yy0, buf, COPY_PUT);
    farfree(buf);
}

void save_job_data(char *ff, dat_t dat)

```

```

{
    FILE *f;

    f = fopen(ff, "wt");
    fwrite(&dat, sizeof(dat), 1, f);
    fclose(f);
}

// symmetric ground fault current
float sym_ground_fault_current(dat_t dat)
{
    return sqrt(norm(3 * dat.ln2ln_voltage / sqrt(3) / (2 * dat.z1 +
dat.z0)));
}

float reduction_factor(dat_t dat)
{
    return (dat.K * 0.63 / dat.K);
//return 0.84;
//return 0.5;
}

float required_conductor_size(dat_t dat, tab_t *tab)
{
    return (dat.i_f * sqrt(((10000 * dat.tf * tab[dat.material].alpha_r *
    tab[dat.material].rho_r / tab[dat.material].TCAP) /
    (log(1 + (dat.Tm - dat.Ta) / (tab[dat.material].K + dat.Ta)))) /
    1000);
}

float reflection_factor(dat_t dat)
{
    return ((dat.rho1 - dat.rho_s) / (dat.rho1 + dat.rho_s));
}

float E_step_50(dat_t dat)
{
    if (dat.hs == 0)
        return ((1000 + 6 * dat.rho1) * 0.157 / sqrt(dat.ts));
    // return ((1000 + 6 * dat.rho1) * 0.116 / sqrt(dat.ts));
    else
        // return ((1000 + 6 * dat.Cs * dat.rho_s) * 0.116 / sqrt(dat.ts));
        return ((1000 + 6 * dat.Cs * dat.rho_s) * 0.157 / sqrt(dat.ts));
}

```

```

float E_touch_50(dat_t dat)
{
    if (dat.hs == 0)
    // return ((1000 + 1.5 * dat.rho1) * 0.116 / sqrt(dat.ts));
    return ((1000 + 1.5 * dat.rho1) * 0.157 / sqrt(dat.ts));
    else
    // return ((1000 + 1.5 * dat.Cs * dat.rho_s) * 0.116 / sqrt(dat.ts));
    return ((1000 + 1.5 * dat.Cs * dat.rho_s) * 0.157 / sqrt(dat.ts));
}

float grid_resistance(dat_t dat)
{
    float r_grid;

    r_grid = dat.rho1 * (1 / (dat.L + /*dat.rod_no */ dat.rod_len) +
        1 / sqrt(20 * dat.A) * (1 + 1 / (1 + dat.h * sqrt(20 / dat.A))));

    /*cout << "dat.L = " << dat.L << "\n";
    cout << "dat.rod_no = " << dat.rod_no << "\n";
    cout << "dat.rod_len = " << dat.rod_len << "\n";
    cout << "dat.A = " << dat.A << "\n";
    cout << "dat.h = " << dat.h << "\n";
    cout << "dat.D = " << dat.D << "\n";
    cout << "dat.n = " << dat.n << "\n";
    cout << "dat.Rg = " << r_grid << "\n";
    getch();*/
    return r_grid;
}

float maximum_grid_current(dat_t dat)
{
    return (dat.Sf * dat.i_f * dat.Df);
}

float decrement_factor(dat_t dat)
{
    float Df;

    if (dat.tf < 0.008)
        Df = 1.65;
    else if (dat.tf < 0.1)
        Df = 1.25;
    else if (dat.tf < 0.25)
        Df = 1.10;
}

```

```

    else
        Df = 1.0;
    return Df;
}

float mesh_voltage(dat_t dat)
{
    float Km, Ki, Kii, Kh;

    if (dat.rod_len > 0)
        Kii = 1;
    else
        Kii = 1 / pow(2 * dat.n, 2 / dat.n);
    Kh = sqrt(1 + dat.h);
    Km = 1 / (2 * M_PI) * (log(dat.D * dat.D / (16 * dat.h * dat.d_min /
1000)) +
        pow(dat.D + 2 * dat.h, 2) / (8 * dat.D * dat.d_min / 1000) -
        dat.h / (4 * dat.d_min / 1000)) + Kii / Kh * log(8 / (M_PI * (2 *
dat.n - 1))));
    Ki = 0.656 + 0.172 * dat.n;

    /*cout << "Kii = " << Kii << "\n";
    cout << "Kh = " << Kh << "\n";
    cout << "Km = " << Km << "\n";
    cout << "Ki = " << Ki << "\n";
    cout << "d = " << dat.d_min << "\n";
    getch();*/

    return (dat.rhol * dat.i_g * Km * Ki / (dat.L + 1.15 * dat.rod_len));
}

void show_result(dat_t dat)
{
    void far *buf1, far *buf2;
    int x0, y0, x1, y1;
    int xx0 = 13, yy0 = 5, xx1 = 67, yy1 = 26;

    x0 = x2px(xx0);
    y0 = y2py(yy0);
    x1 = x2px(xx1);
    y1 = y2py(yy1);
    buf1 = farmalloc(imagesize(x0, y0, x1, y2py(yy0 + 10)));
    buf2 = farmalloc(imagesize(x0, y2py(yy0 + 10), x1, y1));
    getimage(x0, y0, x1, y2py(yy0 + 10), buf1);
    getimage(x0, y2py(yy0 + 10), x1, y1, buf2);

```

```

draw_3d_plate(x0, y0, x1, y1);
hsetbkcolor(DARKGRAY);
hsetcolor(LIGHTCYAN);
hgotoxy(xx0 + 12, yy0 + 1);
hputs("<< FINAL COMPUTATION RESULT >>");
hsetcolor(YELLOW);
hgotoxy(xx0 + 2, yy0 + 3);
hprintf("A) Symmetrical ground fault current [A] : %7.2f", dat.i_f);
hgotoxy(xx0 + 2, yy0 + 4);
hprintf("B) Conductor cross sectional area [mm^2]: %7.2f", dat.Ac);
hgotoxy(xx0 + 2, yy0 + 5);
hprintf("C) Required conductor diameter      [mm] : %7.2f", dat.d_min);
hgotoxy(xx0 + 2, yy0 + 6);
hprintf("D) Reflection factor                      : %7.2f", dat.K);
hgotoxy(xx0 + 2, yy0 + 7);
hprintf("E) Reduction factor                      : %7.2f", dat.Cs);
hgotoxy(xx0 + 2, yy0 + 8);
hprintf("F) Step Voltage Criteria (70)          [V] : %7.2f", dat.Es50);
hgotoxy(xx0 + 2, yy0 + 9);
hprintf("G) Touch Voltage Criteria (70)         [V] : %7.2f", dat.Et50);
hgotoxy(xx0 + 2, yy0 + 10);
hprintf("H) Grid resistance                        [ohm] : %7.2f", dat.Rg);
hgotoxy(xx0 + 2, yy0 + 11);
hprintf("I) Decrement factor                      : %7.2f", dat.Df);
hgotoxy(xx0 + 2, yy0 + 12);
hprintf("J) Maximum grid current                  [A] : %7.2f", dat.i_g);
hgotoxy(xx0 + 2, yy0 + 13);
hprintf("K) Ground Potential Rise                [V] : %7.2f", dat.i_g * dat.Rg);
hgotoxy(xx0 + 2, yy0 + 14);
hprintf("L) Mesh voltage                        [V] : %7.2f", dat.Em);
hgotoxy(xx0 + 2, yy0 + 15);
hprintf("M) Spacing between conductors          [m] : %7.2f", dat.D);
hgotoxy(xx0 + 2, yy0 + 16);
hsetcolor(LIGHTCYAN);
hrestartcursor();
hgotoxy(xx0 + 2, yy1 - 2);
hputs("아무 키나 누르십시오 ....");
hgetch();
putimage(x0, y0, buf1, COPY_PUT);
putimage(x0, y2py(yy0 + 10), buf2, COPY_PUT);
farfree(buf1);
farfree(buf2);
}

```

```

void grounding_system_design(dat_t &dat, tab_t *tab, char *ff)
{
    char    satisfy_criteria;
    void    far *buf;
    float    tmp;
    int      x0, y0, x1, y1;
    int      xx0 = 16, yy0 = 13, xx1 = 65, yy1 = 22;

    if (!strcmp(ff, NO_SELECT))    {
        no_job_error();
        return;
    }
    //-----
    // STEP 1                field data
    //-----
    dat.A = dat.Ax * dat.Ay;
    //soil resistivity

    //-----
    // STEP 2                conductor size
    //-----
    if (dat.i_f == 0)
        dat.i_f = sym_ground_fault_current(dat);
    dat.Ac = required_conductor_size(dat, tab);
    dat.d_min = sqrt(dat.Ac * 4 / M_PI);

    x0 = x2px(xx0);
    y0 = y2py(yy0);
    x1 = x2px(xx1);
    y1 = y2py(yy1);
    buf = farmalloc(imagesize(x0, y0, x1, y1));
    getimage(x0, y0, x1, y1, buf);
    draw_3d_plate(x0, y0, x1, y1);
    hsetbkcolor(DARKGRAY);
    hsetcolor(LIGHTCYAN);
    hgotoxy(xx0 + 9, yy0 + 1);
    hputs("<<< COMPUTED CONDUCTOR SIZE >>>");
    hsetcolor(YELLOW);
    hgotoxy(xx0 + 2, yy0 + 2);
    hprintf("A) Required conductor size      = %6.2f [mm^2]", dat.Ac);
    hgotoxy(xx0 + 2, yy0 + 3);
    hprintf("B) minimum conductor diameter = %6.2f [mm]", dat.d_min);
    hsetcolor(LIGHTCYAN);
    hgotoxy(xx0 + 2, yy0 + 5);
    hputs("Please enter the following value ...");
}

```

```

hsetcolor(YELLOW);
hgotoxy(xx0 + 2, yy0 + 6);
hputs("a) Designed grid diameter size [mm] = ");
tmp = get_data(xx0 + 42, yy0 + 6, "#999.9");
if (tmp > 0)
    dat.d_min = tmp;
//dat.d_min = sqrt(dat.Ac * 4 / M_PI);
putimage(x0, y0, buf, COPY_PUT);
farfree(buf);
//-----
// STEP 3                touch and step criteria
//-----
dat.K = reflection_factor(dat);
if (dat.hs > 0)
    dat.Cs = reduction_factor(dat);
else
    dat.Cs = 1;
dat.Es50 = E_step_50(dat);
dat.Et50 = E_touch_50(dat);

y1 = y2py(yy1 - 2);
buf = farmalloc(imagesize(x0, y0, x1, y1));
getimage(x0, y0, x1, y1, buf);
draw_3d_plate(x0, y0, x1, y1);
hsetcolor(LIGHTCYAN);
hgotoxy(xx0 + 6, yy0 + 1);
hputs("<<< STEP & TOUCH VOLTAGE CRITERIA >>>");
hsetcolor(YELLOW);
hgotoxy(xx0 + 2, yy0 + 2);
hprintf("A) STEP VOLTAGE 70    : %7.2f [V]", dat.Es50);
hgotoxy(xx0 + 2, yy0 + 3);
hprintf("B) TOUCH VOLTAGE 70   : %7.2f [V]", dat.Et50);
hsetcolor(LIGHTCYAN);
hgotoxy(xx0 + 2, yy0 + 5);
hputs("Please strike any key to continue ...");
hgetch();
putimage(x0, y0, buf, COPY_PUT);
farfree(buf);
//-----
// STEP 4                initial design
//-----
dat.n = ceil(sqrt(dat.A) / dat.D + 0.5);
dat.L = 2 * dat.n * sqrt(dat.A);

//-----

```

```

// STEP 5                determination of grid resistance
//-----
dat.Rg = grid_resistance(dat);
//y1 = y2py(yy1 - 2);
buf = farmalloc(imagesize(x0, y0, x1, y1));
getimage(x0, y0, x1, y1, buf);
draw_3d_plate(x0, y0, x1, y1);
hsetcolor(LIGHTCYAN);
hgotoxy(xx0 + 13, yy0 + 1);
hputs("<<< GRID RESISTANCE >>>");
hsetcolor(YELLOW);
hgotoxy(xx0 + 2, yy0 + 2);
hprintf("A) Required grid resist. : %7.2f [ohm]", dat.r_rg);
hgotoxy(xx0 + 2, yy0 + 3);
hprintf("A) GRID RESISTANCE          : %7.2f [ohm]", dat.Rg);
hsetcolor(LIGHTCYAN);
hgotoxy(xx0 + 2, yy0 + 5);
/*      if (dat.Rg > dat.r_rg)
        hputs("Rg < Rreq, please re-enter the data ...");
else*/
hputs("Strike any key to continue ...");
hgetch();
putimage(x0, y0, buf, COPY_PUT);
farfree(buf);

//-----
// STEP 6                maximum grid current
//-----
dat.Df = decrement_factor(dat);
dat.i_g = maximum_grid_current(dat);

//-----
// STEP 7                Ground Potential Rise
//-----
// GPR = dat.i_g * dat.Rg;

//-----
// STEP 8                mesh voltage
//-----
dat.Em = mesh_voltage(dat);

//-----
// STEP 9                modify initial design
//-----

```

```

satisfy_criteria = 1;
y1 = y2py(yy1 + 1);
buf = farmalloc(imagesize(x0, y0, x1, y1));
getimage(x0, y0, x1, y1, buf);
draw_3d_plate(x0, y0, x1, y1);
hsetcolor(LIGHTCYAN);
hgotoxy(xx0 + 6, yy0 + 1);
hputs("<<< COMPUTATION RESULTS EVALUATION >>>");
hsetcolor(YELLOW);
hgotoxy(xx0 + 2, yy0 + 2);
hputs("----- 기준값 -- 계산값 -----");
if (dat.Et50 > dat.Em)
    hsetcolor(LIGHTGREEN);
else
    {
        sound(500);
        delay(200);
        nosound();
        satisfy_criteria = 0;
        hsetcolor(LIGHTRED);
    }
hgotoxy(xx0 + 2, yy0 + 3);
hprintf("A) TOUCH VOLTAGE : %7.2f ", dat.Et50);
hsetcolor(WHITE);
if (dat.Et50 > dat.Em)
    hprintf(">"),
    hsetcolor(LIGHTGREEN);
else
    hprintf("<"),
    hsetcolor(LIGHTRED);
hprintf(" %7.2f [V]", dat.Em);
if (dat.r_rg > dat.Rg)
    hsetcolor(LIGHTGREEN);
else
    {
        sound(500);
        delay(200);
        nosound();
        satisfy_criteria = 0;
        hsetcolor(LIGHTRED);
    }
hgotoxy(xx0 + 2, yy0 + 4);
hprintf("B) GRID RESISTANCE : %7.2f ", dat.r_rg);
hsetcolor(WHITE);
if (dat.r_rg > dat.Rg)
    hprintf(">"),

```

```

        hsetcolor(LIGHTGREEN);
    else
        hprintf("<"),
        hsetcolor(LIGHTRED);
    hprintf(" %7.2f [ohm]", dat.Rg);
    hsetcolor(YELLOW);
    hgotoxy(xx0 + 2, yy1 - 3);
    if (satisfy_criteria) {
        hputs("기준값을 만족하는 설계입니다.");
    }
    else {
        hputs("기준값을 만족시키지 못하는 설계입니다.");
        hgotoxy(xx0 + 2, yy1 - 2);
        hputs("접지제원을 수정하십시오.");
    }
    hsetcolor(LIGHTCYAN);
    hgotoxy(xx0 + 2, yy1 - 1);
    hputs("아무 키나 치면 처음으로 되돌아갑니다 ....");
    hgetch();
    putimage(x0, y0, buf, COPY_PUT);
    farfree(buf);
}

void userinithan(void)
{
    LINK_VGA_hputfont8nx16m();
    LINK_VGA_hputfont16nx16m();
    inithanlib(BGIMODE, HANVGA, HAN212MD1, ENG5D5);
    if (hgraphresult() < NOERROR) {
        cerr << "\x7" << "inithanlib() function error";
        exit(0);
    }
    SELECT_hputfont8nx16m(AUTOSELECT);
    SELECT_hputfont16nx16m(AUTOSELECT);
}

void logo(void)
{
    setbkcolor(BLUE);
    clearviewport();
    hsetcolor(LIGHTCYAN);
    hsetoutputmode(OVERWRITE_ENLARGE);
    hsetenlarge(3, 4);
    hgotopxy(x2px(2) + 12, y2py(2));

```

```

hputs("GROUNDING SYSTEM DESIGN");
hsetenlarge(1, 2);
hsetcolor(LIGHTGRAY);
hgotoxy(18, 11);
hputs("한국산업안전공단 산업안전연구원 기계전기연구실");
hsetenlarge(1, 1);
hgotoxy(31, 10);
hsetcolor(LIGHTRED);
hputs("Ver. 1.0 (C) 1994");
hgotoxy(15, 24);
hsetcolor(LIGHTGRAY);
hputs("Tel. (032) 513-0230, 518-6484/6, Fax. (032) 518-6483");
hsetcolor(LIGHTGREEN);
hgotoxy(27, 27);
hputs("아무 키나 누르십시오...");
hrestartcursor();
hgetch();
hgotoxy(27, 27);
hputs("                ");
}

void read_table_data(char *ff, tab_t *tab)
{
    FILE *f;
    int i, j, k;

    if (!(f = fopen(ff, "rt"))) {
        cout << "\x7" << "File " << ff << " not opened !";
        exit(0);
    }
    for (i = 0; i < TAB_NO; i++) {
        fscanf(f, "%s", tab[i].descript);
        for (j = 0, k = strlen(tab[i].descript); j < k; j++)
            if (tab[i].descript[j] == '*')
                tab[i].descript[j] = ' ';
        tab[i].descript[j] = '\0';
        fscanf(f, "%f %f %f %f %f %f", &tab[i].mc, &tab[i].alpha_r,
            &tab[i].K, &tab[i].Tf, &tab[i].rho_r, &tab[i].TCAP);
    }
    fclose(f);
}

int select_material(tab_t *tab, dat_t dat)
{
    int pos, pos0, _x0, _y0, _x1, _y1, i;

```

```

int      xx0 = 13, yy0 = 8, xx1 = 68, yy1 = 26;
char     material_selected, ch;
void far *buf;

_x0 = x2px(xx0 + 12);
_y0 = y2py(yy0 + 2);
_x1 = x2px(xx1 - 12);
_y1 = y2py(yy1 - 3);
buf = farmalloc(imagesize(_x0, _y0, _x1, _y1));
getimage(_x0, _y0, _x1, _y1, buf);
draw_3d_plate(_x0, _y0, _x1, _y1);
pos = dat.material;
hsetbkcolor(DARKGRAY);
hsetcolor(LIGHTCYAN);
hgotoxy(xx0 + 14, yy0 + 3);
hputs("전선재료를 선택해주십시오...");
hsetcolor(YELLOW);
for (i = 0; i < TAB_NO; i++) {
    if (pos == i)
        hsetbkcolor(BLUE);
    else
        hsetbkcolor(DARKGRAY);
    hgotoxy(xx0 + 14, yy0 + 4 + i);
    hputs(tab[i].descript);
}
material_selected = 0;
do {
    pos0 = pos;
    if ((ch = getch()) == 0)
        ch = getch();
    switch (ch) {
        case 72:
            pos--;
            break;
        case 80:
            pos++;
            break;
        case 71:
        case 73:
            pos = 0;
            break;
        case 79:
        case 81:
            pos = TAB_NO - 1;
            break;
    }
} while (material_selected != 1);

```

```

        case 13:
            material_selected = 1;
            break;
        default:
            break;
    }
    if (pos < 0)
        pos += TAB_NO;
    else if (pos > TAB_NO - 1)
        pos -= TAB_NO;
    hsetbkcolor(DARKGRAY);
    hgotoxy(xx0 + 14, yy0 + 4 + pos0);
    hputs(tab[pos0].descript);
    hsetbkcolor(BLUE);
    hgotoxy(xx0 + 14, yy0 + 4 + pos);
    hputs(tab[pos].descript);
} while (!material_selected);
putimage(_x0, _y0, buf, COPY_PUT);
farfree(buf);
hsetbkcolor(DARKGRAY);
hsetcolor(YELLOW);

return pos;
}

void create_new_job(dat_t &dat, tab_t *tab, char *ff)
{
    void far *buf1, far *buf2;
    double r, i;
    char ch, fn[13], buffer[MLEN + 1];
    FILE *f;
    int x0, y0, x1, y1, pos;
    int xx0 = 13, yy0 = 4, xx1 = 68, yy1 = 28;

    x0 = x2px(xx0);
    y0 = y2py(yy0);
    x1 = x2px(xx1);
    y1 = y2py(yy1);
    buf1 = farmalloc(imagesize(x0, y0, x1, y2py(yy0 + 9)));
    buf2 = farmalloc(imagesize(x0, y2py(yy0 + 9), x1, y1));
    getimage(x0, y0, x1, y2py(yy0 + 9), buf1);
    getimage(x0, y2py(yy0 + 9), x1, y1, buf2);
    draw_3d_plate(x0, y0, x1, y1);
    hsetcolor(LIGHTCYAN);
    hsetbkcolor(DARKGRAY);

```

```

hgotoxy(xx0 + 2, yy0 + 1);
hputs("다음의 데이터를 입력해 주십시오 ...");
hsetcolor(YELLOW);
hgotoxy(xx0 + 2, yy0 + 2);
hputs("A) Job Name           [8자 이내] :");
hgotoxy(xx0 + 2, yy0 + 3);
hputs("B) Required ground resist. [ohm] :");
hgotoxy(xx0 + 2, yy0 + 4);
hputs("C) Duration of shock      [sec] :");
hgotoxy(xx0 + 2, yy0 + 5);
hputs("D) Fault duration        [sec] :");
hgotoxy(xx0 + 2, yy0 + 6);
hputs("E) Fault impendace 1      [ohm] :      +j");
hgotoxy(xx0 + 2, yy0 + 7);
hputs("E) Fault impedance 0     [ohm] :      +j");
hgotoxy(xx0 + 2, yy0 + 8);
hputs("F) Sym. Fault Current    [A] :");
hgotoxy(xx0 + 2, yy0 + 9);
hputs("G) Line to line voltage  [V] :");
hgotoxy(xx0 + 2, yy0 + 10);
hputs("H) Current division factor :");
hgotoxy(xx0 + 2, yy0 + 11);
hputs("I) Thickness of crushed rock [m] :");
hgotoxy(xx0 + 2, yy0 + 12);
hputs("J) Crushed rock resistivity [ohm-m]:");
hgotoxy(xx0 + 2, yy0 + 13);
hputs("K) Soil resistivity      [ohm-m] :");
hgotoxy(xx0 + 2, yy0 + 14);
hputs("L) Depth of grid burial   [m] :");
hgotoxy(xx0 + 2, yy0 + 15);
hputs("M) Grounding area        [m x m] :      x");
hgotoxy(xx0 + 2, yy0 + 16);
hputs("N) Max.allowable temperature['C] :");
hgotoxy(xx0 + 2, yy0 + 17);
hputs("O) Ambient temperature   ['C] :");
hgotoxy(xx0 + 2, yy0 + 18);
hputs("P) Grounding material     :");
hgotoxy(xx0 + 2, yy0 + 19);
hputs("Q) Spacing between conductors [m]:");
hgotoxy(xx0 + 2, yy0 + 20);
hputs("R) A ground rod length    [m] :");
/* hgotoxy(xx0 + 2, yy0 + 21);
hputs("S) no. of ground rod      :");*/

hrestartcursor();

```

```

_hangulmode = false;

memset(buffer, ' ', MLEN);
buffer[MLEN + 1] = '\0';
do {
    hgotoxy(xx0 + 37, yy0 + 2);
    hgetdata(buffer, "!!!!!!");
} while (!isalpha(buffer[0]));
strcpy(dat.jobid, buffer);
dat.tf = dat.ln2ln_voltage = dat.Sf = dat.hs = dat.rho_s = 0;
dat.rho1 = dat.h = dat.Ax = dat.Ay = dat.Tm = dat.Ta = dat.i_f = 0;
dat.z1 = dat.z0 = complex(0, 0);
dat.r_rg = get_data(xx0 + 37, yy0 + 3, "#999.9");
dat.ts = get_data(xx0 + 37, yy0 + 4, "#999.9");
dat.tf = get_data(xx0 + 37, yy0 + 5, "#999.9");
r = get_data(xx0 + 37, yy0 + 6, "#999.9");
i = get_data(xx0 + 47, yy0 + 6, "#999.9");
dat.z1 = complex(r, i);
r = get_data(xx0 + 37, yy0 + 7, "#999.9");
i = get_data(xx0 + 47, yy0 + 7, "#999.9");
dat.z0 = complex(r, i);
dat.i_f = get_data(xx0 + 37, yy0 + 8, "#99999.9");
dat.ln2ln_voltage = get_data(xx0 + 36, yy0 + 9, "9999999");
dat.Sf = get_data(xx0 + 37, yy0 + 10, "#999.9");
dat.hs = get_data(xx0 + 37, yy0 + 11, "#999.9");
if (dat.hs == 0)
    dat.rho_s = 0;
else
    dat.rho_s = get_data(xx0 + 39, yy0 + 12, "#9999.9");
dat.rho1 = get_data(xx0 + 37, yy0 + 13, "#999.9");
dat.h = get_data(xx0 + 37, yy0 + 14, "#999.9");
dat.Ax = get_data(xx0 + 37, yy0 + 15, "#99999");
dat.Ay = get_data(xx0 + 47, yy0 + 15, "#99999");
dat.Tm = get_data(xx0 + 37, yy0 + 16, "#999.9");
dat.Ta = get_data(xx0 + 37, yy0 + 17, "#999.9");
dat.material = 0;
pos = select_material(tab, dat);
hgotoxy(xx0 + 35, yy0 + 18);
hputs(tab[pos].descript);
dat.material = pos;
dat.D = get_data(xx0 + 37, yy0 + 19, "#999.9");
dat.rod_len = get_data(xx0 + 37, yy0 + 20, "#999.9");
// dat.rod_no = get_data(xx0 + 37, yy0 + 21, "#99999");

hsetcolor(LIGHTCYAN);

```

```

hsetbkcolor(DARKGRAY);
hgotoxy(15, yy1 - 2);
hputs("입력한 데이터가 모두 맞습니까 [Y/N] ? ");
do {
    ch = toupper(hgetch());
} while (ch != 'Y' && ch != 'N');
if (ch == 'Y') {
    strcpy(fn, dat.jobid);
    strcpy(&fn[strlen(fn)], ".JOB");
    f = fopen(fn, "rt");
    if (!f) {
        save_job_data(fn, dat);
        strcpy(ff, fn);
    }
    else {
        hgotoxy(15, yy1 - 2);
        hputs("JOB이 이미 존재합니다. 겹쳐쓸까요 [Y/N] ? ");

        do {
            ch = toupper(hgetch());
        } while (ch != 'Y' && ch != 'N');
        if (ch == 'Y') {
            f = fopen(fn, "wt");
            fwrite(&dat, sizeof(dat), 1, f);
            strcpy(ff, fn);
        }
        fclose(f);
    }
}

putimage(x0, y0, buf1, COPY_PUT);
putimage(x0, y2py(yy0 + 9), buf2, COPY_PUT);
farfree(buf1);
farfree(buf2);
}

void view_job(dat_t dat, tab_t *tab, char *ff)
{
    void far *buf1, far *buf2;
    int x0, y0, x1, y1;
    int xx0 = 13, yy0 = 5, xx1 = 68, yy1 = 27;

    if (!strcmp(ff, NO_SELECT)) {
        no_job_error();
        return;
    }
}

```

```

y0 = y2py(yy0);
x1 = x2px(xx1);
y1 = y2py(yy1);
buf1 = farmalloc(imagesize(x0, y0, x1, y2py(yy0 + 9)));
buf2 = farmalloc(imagesize(x0, y2py(yy0 + 9), x1, y1));
getimage(x0, y0, x1, y2py(yy0 + 9), buf1);
getimage(x0, y2py(yy0 + 9), x1, y1, buf2);
draw_3d_plate(x0, y0, x1, y1);
hsetbkcolor(DARKGRAY);
hsetcolor(YELLOW);
hgotoxy(xx0 + 2, yy0 + 1);
hprintf("A) Job Name           [8자 이내] : %-8s", dat.jobid);
hgotoxy(xx0 + 2, yy0 + 2);
hprintf("B) Required ground resist. [ohm] : %5.1f", dat.r_rg);
hgotoxy(xx0 + 2, yy0 + 3);
hprintf("C) Duration of shock      [sec] : %5.1f", dat.ts);
hgotoxy(xx0 + 2, yy0 + 4);
hprintf("D) Fault duration         [sec] : %5.1f", dat.tf);
hgotoxy(xx0 + 2, yy0 + 5);
hprintf("E) Fault impendace 1      [ohm] : %5.1f +j %5.1f",
      real(dat.z1), imag(dat.z1));
hgotoxy(xx0 + 2, yy0 + 6);
hprintf("F) Fault impedance 0     [ohm] : %5.1f +j %5.1f",
      real(dat.z0), imag(dat.z0));
hgotoxy(xx0 + 2, yy0 + 7);
hprintf("G) Sym. Fault Current      [A] : %8.0f", dat.i_f);
hgotoxy(xx0 + 2, yy0 + 8);
hprintf("H) Line to line voltage    [V] : %8.0f", dat.ln2ln_voltage);
hgotoxy(xx0 + 2, yy0 + 9);
hprintf("I) Current division factor : %5.1f", dat.Sf);
hgotoxy(xx0 + 2, yy0 + 10);
hprintf("J) Thickness of crushed rock [m] : %5.1f", dat.hs);
hgotoxy(xx0 + 2, yy0 + 11);
hprintf("K) Crushed rock resistivity [ohm-m]: %5.1f", dat.rho_s);
hgotoxy(xx0 + 2, yy0 + 12);
hprintf("L) Soil resistivity        [ohm-m] : %5.1f", dat.rho1);
hgotoxy(xx0 + 2, yy0 + 13);
hprintf("M) Depth of grid burial     [m] : %5.1f", dat.h);
hgotoxy(xx0 + 2, yy0 + 14);
hprintf("N) Grounding area          [m x m] : %5.1f x %5.1f", dat.Ax,
dat.Ay);
hgotoxy(xx0 + 2, yy0 + 15);
hprintf("O) Max.allowable temperature['C] : %5.1f", dat.Tm);
hgotoxy(xx0 + 2, yy0 + 16);
hprintf("P) Ambient temperature      ['C] : %5.1f", dat.Ta);

```

```

        hgotoxy(xx0 + 2, yy0 + 17);
        hprintf("Q) Grounding material : %17s",
tab[dat.material].descrip);
        hgotoxy(xx0 + 2, yy0 + 18);
        hprintf("R) Initial spacing [m] : %5.1f", dat.D);
        hgotoxy(xx0 + 2, yy0 + 19);
        hprintf("S) Ground rod length [m] : %5.1f", dat.rod_len);
        hsetcolor(LIGHTCYAN);
        hgotoxy(xx0 + 2, yy1 - 2);
        hputs("아무 키나 누르세요 ....");
        hgetch();
        putimage(x0, y0, buf1, COPY_PUT);
        putimage(x0, y2py(yy0 + 9), buf2, COPY_PUT);
        farfree(buf1);
        farfree(buf2);
}

```

```

void load_job(dat_t &dat, tab_t *tab, char *ff)
{
    struct ffbk ffbk;
    void far *buf;
    unsigned size;
    FILE *f;
    int pos, pos0, i, done, k, x0 = 49, y0 = 10;
    int xx0, yy0, xx1, yy1;
    char fitem[MAX_FILE_NO][9], *ptr, ch, job_selected, fn[13];

    k = 0;
    dat.rod_no = 0;
    done = findfirst("*.job", &ffbk, FA_ARCH);
    if (!done) {
        memset(fitem[k], ' ', 8);
        fitem[k][8] = '\0';
        ptr = (char *)memchr(ffbk.ff_name, '.', strlen(ffbk.ff_name));
        strncpy(fitem[k++], ffbk.ff_name, ptr - ffbk.ff_name);
        while (!done)
        {
            done = findnext(&ffbk);
            if (!done) {
                memset(fitem[k], ' ', 8);
                fitem[k][8] = '\0';
                ptr=(char*)memchr(ffbk.ff_name, '.',strlen(ffbk.ff_name));
                strncpy(fitem[k++], ffbk.ff_name, ptr - ffbk.ff_name);
            }
            if (k >= MAX_FILE_NO)

```

```

        break;
    }
}
else
    return;
xx0 = x2px(x0);
yy0 = y2py(y0);
xx1 = x2px(x0 + 10);
yy1 = y2py(y0 + k + 2);
size = imagesize(xx0, yy0, xx1, yy1);
buf = farmalloc(size);
getimage(xx0, yy0, xx1, yy1, buf);
draw_3d_plate(xx0, yy0, xx1, y2py(y0 + k + 2));

pos = 0;
for (i = 0; i < k; i++) {
    if (pos == i) {
        hsetbkcolor(BLUE);
        hsetcolor(YELLOW);
    }
    else {
        hsetbkcolor(DARKGRAY);
        hsetcolor(YELLOW);
    }
    hgotoxy(x0 + 1, y0 + i + 1);
    hputs(fitem[i]);
}

job_selected = 0;
do {
    ch = getch();
    if (ch == 0)
        ch = getch();
    pos0 = pos;
    switch (ch) {
        case 13:
            job_selected = 1;
            break;
        case 27:
            putimage(xx0, yy0, buf, COPY_PUT);
            farfree(buf);
            return;
        case 71:
        case 73:
            pos = 0;

```

```

        break;
    case 79:
    case 81:
        pos = k - 1;
        break;
    case 72:
        pos--;
        break;
    case 80:
        pos++;
        break;
    default:
        break;
}
if (pos < 0)
    pos += k;
else if (pos > k - 1)
    pos -= k;
hsetbkcolor(DARKGRAY);
hsetcolor(YELLOW);
hgotoxy(x0 + 1, y0 + pos0 + 1);
hputs(fitem[pos0]);
for (i = strlen(fitem[pos0]); i < 8; i++)
    hputs(" ");
hsetbkcolor(BLUE);
hsetcolor(YELLOW);
hgotoxy(x0 + 1, y0 + pos + 1);
hputs(fitem[pos]);
for (i = strlen(fitem[pos]); i < 8; i++)
    hputs(" ");
} while (!job_selected);
strcpy(fn, fitem[pos]);
strcpy(&fn[strlen(fn)], ".job");
dat.tf = dat.ln2ln_voltage = dat.Sf = dat.hs = dat.rho_s = 0;
dat.rho1 = dat.h = dat.Ax = dat.Ay = dat.Tm = dat.Ta = dat.i_f = 0;
dat.z1 = dat.z0 = complex(0, 0);
f = fopen(fn, "rt");
fread(&dat, sizeof(dat), 1, f);
fclose(f);
strcpy(ff, fn);
putimage(xx0, yy0, buf, COPY_PUT);
farfree(buf);
view_job(dat, tab, ff);
}

```

```

void view_3d_graph(char *ff)
{
    void    far *buf;
    plot3dg(ff, X0, Y0, X1, Y1);

    buf = farmalloc(imagesize(X0 - 5, Y0 - 5, X1 + 7, Y1 + 7));
    if (!buf) {
        cout << "x7" << "buf-memory allocation error";
        exit(0);
    }
    getimage(X0 - 5, Y0 - 5, X1 + 7, Y1 + 7, buf);
    draw_3d_plate(X0 - 5, Y0 - 5, X1 + 5, Y1 + 5);
    setcolor(YELLOW);
    g.plot_3d_coord();
    setcolor(YELLOW);
    g.space3d();
    getch();
    putimage(X0 - 5, Y0 - 5, buf, COPY_PUT);
    farfree(buf);
}

void calculate_rho(dat_t &dat, char *ff)
{
    FILE *f, *fb;
    char fd[13];
    int  a, b, c, d;
    float e, h, i, j, tmp;
    int  x0, y0, x1, y1;
    void    far *buf;

    if (!strcmp(ff, NO_SELECT)) {
        no_job_error();
        return;
    }

    /*x0 = x2px(16);
    y0 = y2py(5);
    x1 = x2px(63);
    y1 = y2py(27);
    buf = farmalloc(imagesize(x0, y0, x1, y1));
    getimage(x0, y0, x1, y1, buf);
    draw_3d_plate(x0, y0, x1, y1);
    hsetbkcolor(DARKGRAY);
    hsetcolor(LIGHTCYAN);
    hgotoxy(20, 6);

```

```

hputs("측정 횟수   측정 간격 (m)   측정 값 (ohm)");*/

strcpy(fd, ff);
strcpy(&fd[9], "rho");
f = fopen("res_rho.dat", "wt");
fb = fopen(fd, "rt");
fscanf(fb, "%d %d %d %d", &a, &b, &c, &d);
fscanf(fb, "%f %f %f %f", &e, &h, &i, &j);
fprintf(f, "%5d\n%5d\n%5d\n%5d\n", a, b, c, d);
fprintf(f, "%9.5f\n%8.2f %9.2f %8.2f\n", e, h, i, j);
fscanf(fb, "%f %f %f", &e, &h, &i);
fprintf(f, "%6.1f %10.1f %8.1f\n", e, h, i);
hsetbkcolor(DARKGRAY);
hsetcolor(YELLOW);
for (d = 0; d < b; d++) {
    fscanf(fb, "%d %f %f %f", &c, &e, &h, &i);
    fprintf(f, "%5d %8.4f %8.3f %8.3f\n", c, e, h, i);
    // hgotoxy(22, d + 7);
    // hprintf("%2d) %12.1f %15.1f\n", d + 1, e, h);
}
fclose(f);
fclose(fb);

/* hsetcolor(LIGHTCYAN);
hrestartcursor();
hgotoxy(18, 26);
hputs("아무 키나 치십시오 ....");
hgetch();
// if (toupper(getch()) == 'Q');
//     return;
// return;
putimage(x0, y0, buf, COPY_PUT);
farfree(buf);    */

x0 = x2px(16);
y0 = y2py(15);
x1 = x2px(63);
y1 = y2py(25);
buf = farmalloc(imagesize(x0, y0, x1, y1));
getimage(x0, y0, x1, y1, buf);
draw_3d_plate(x0, y0, x1, y0 + y2py(4));
hsetbkcolor(DARKGRAY);
hsetcolor(YELLOW);
hgotoxy(22, 16);
hputs("Please wait while calculating ...");

```

```

spawnl(P_WAIT, "rhom.exe", NULL);
draw_3d_plate(x0, y0, x1, y1);
f = fopen("last_rho.out", "rt");
fscanf(f, "%f", &tmp);
hsetcolor(LIGHTCYAN);
hgotoxy(20, 16);
hputs("-----<< COMPUTATION RESULTS >>-----");
hsetcolor(YELLOW);
hgotoxy(19, 18);
hprintf("Top Layer Resistivity      : %6.2f [ohm-m]", tmp);
dat.rho1 = tmp;
fscanf(f, "%f", &tmp);
hgotoxy(19, 19);
hprintf("Bottom Layer Resistivity : %6.2f [ohm-m]", tmp);
dat.rho2 = tmp;
fscanf(f, "%f", &tmp);
hgotoxy(19, 20);
hprintf("Reflection Coefficient   : %6.2f [p.u]", tmp);
dat.K = tmp;
fscanf(f, "%f", &tmp);
hgotoxy(19, 21);
hprintf("Top Layer Height           : %6.2f [m]", tmp);
dat.h_tl = tmp;
fclose(f);
save_job_data(ff, dat);
hsetcolor(LIGHTCYAN);
hgotoxy(19, 23);
hputs("아무 키나 누르십시오 ...");
hgetch();
putimage(x2px(16), y2py(15), buf, COPY_PUT);
farfree(buf);
}

```

```

void new_measurement_data(char *ff)
{
    FILE    *f;
    char    ch, fn[13];
    void    far *buf1, far *buf2;
    int      x0, y0, x1, y1, i, measure_no;
    int      xx0 = 13, yy0 = 4, xx1 = 68, yy1 = 26;
    struct tmp_t {
        float    dis;
        float    measure;
    }
    *tmp;
}

```

```

if (!strcmp(ff, NO_SELECT))    {
    no_job_error();
    return;
}
else    {
    strcpy(fn, ff);
    strcpy(&fn[9], "rho");
}
x0 = x2px(xx0);
y0 = y2py(yy0);
x1 = x2px(xx1);
y1 = y2py(yy1);
buf1 = farmalloc(imagesize(x0, y0, x1, y2py(yy0 + 11)));
buf2 = farmalloc(imagesize(x0, y2py(yy0 + 11), x1, y1));
getimage(x0, y0, x1, y2py(yy0 + 11), buf1);
getimage(x0, y2py(yy0 + 11), x1, y1, buf2);
draw_3d_plate(x0, y0, x1, y1);
hsetbkcolor(DARKGRAY);
hsetcolor(LIGHTCYAN);
hgotoxy(xx0 + 2, yy0 + 1);
hputs("대지저항 측정 데이터를 입력하여 주십시오.");
hsetcolor(LIGHTRED);
hgotoxy(xx0 + 2, yy0 + 3);
hputs("측정 횟수 :");
hscanf("%d", &measure_no);
tmp = (struct tmp_t *)malloc(measure_no * sizeof(struct tmp_t));
for (i = 0; i < measure_no; i++)    {
    hgotoxy(xx0 + 2, yy0 + 4 + i);
    hprintf("%2d", i + 1);
}
hgotoxy(xx0 + 17, yy0 + 3);
hputs("측정 간격 [m]");
hgotoxy(xx0 + 35, yy0 + 3);
hputs("측정 값 [ohm]");
hrestartcursor();
hsetcolor(YELLOW);
for (i = 0; i < measure_no; i++)    {
    hgotoxy(xx0 + 19, yy0 + 4 + i);
    hscanf("%f", &tmp[i].dis);
    hgotoxy(xx0 + 37, yy0 + 4 + i);
    hscanf("%f", &tmp[i].measure);
}
hrestartcursor();
hsetcolor(LIGHTCYAN);
hgotoxy(xx0 + 2, yy1 - 2);

```

```

hputs("입력한 데이터가 다 맞습니까 [Y/N] ? ");
do {
    ch = toupper(getch());
} while (ch != 'Y' && ch != 'N');
if (ch == 'Y') {
    f = fopen(fn, "wt");
    fprintf(f, "%5d\n", 0);
    fprintf(f, "%5d\n", measure_no);
    fprintf(f, "%5d\n", 130);
    fprintf(f, "%5d\n", 100);
    fprintf(f, "%9.5f\n", 0.00001);
    fprintf(f, "%8.2f %9.2f %8.2f\n", -0.05, -0.10, -0.20);
    fprintf(f, "%6.1f %10.1f %8.1f\n", 30.2, -0.6, 1.0);
    for (i = 0; i < measure_no; i++)
        fprintf(f, "%5d %8.4f %8.3f %8.3f\n", i + 1,
tmp[i].dis, tmp[i].measure, 0.001);
    fclose(f);
}
putimage(x0, y0, buf1, COPY_PUT);
putimage(x0, y2py(yy0 + 11), buf2, COPY_PUT);
farfree(buf1);
farfree(buf2);
free(tmp);
}

void modify_initial_design(char *ff, dat_t &dat)
{
    int i, x0, y0, x1, y1;
    int xx0 = 13, yy0 = 12, xx1 = 68, yy1 = 21;
    int pos, item_selected;
    float tmp;
    char ch;
    void far *buf;

    if (!strcmp(ff, NO_SELECT)) {
        no_job_error();
        return;
    }
    x0 = x2px(xx0);
    y0 = y2py(yy0);
    x1 = x2px(xx1);
    y1 = y2py(yy1);
    buf = farmalloc(imagesize(x0, y0, x1 + 2, y1 + 2));
    getimage(x0, y0, x1 + 2, y1 + 2, buf);
    draw_3d_plate(x0, y0, x1, y1);

```

```

hsetbkcolor(DARKGRAY);
hsetcolor(LIGHTCYAN);
hgotoxy(xx0 + 2, yy0 + 1);
hputs("수정할 데이터를 선택하십시오.");
hsetcolor(YELLOW);
pos = 0;
for (i = 0; i < 2; i++) {
    hsetbkcolor(BLUE);
    hgotoxy(xx0 + 2, yy0 + 3);
    hprintf("    A) Mesh spacing [m]      : %5.0f    ", dat.D);
    hsetbkcolor(DARKGRAY);
    hgotoxy(xx0 + 2, yy0 + 4);
    hprintf("    B) Ground rod length [m] : %5.0f    ",
dat.rod_len);
}
item_selected = pos = 0;
do {
    ch = getch();
    if (ch == 0)
        ch = getch();
    if (ch == 13)
        item_selected = 1;
    else if (ch == 72)
        pos = 0;
    else if (ch == 80)
        pos = 1;
    else if (ch == 27) {
        putimage(x0, y0, buf, COPY_PUT);
        farfree(buf);
        return;
    }
    if (pos == 0)
        hsetbkcolor(BLUE);
    else
        hsetbkcolor(DARKGRAY);
    hgotoxy(xx0 + 2, yy0 + 3);
    hprintf("    A) Mesh spacing [m]      : %5.0f    ", dat.D);
    if (pos == 1)
        hsetbkcolor(BLUE);
    else
        hsetbkcolor(DARKGRAY);
    hgotoxy(xx0 + 2, yy0 + 4);
    hprintf("    B) Ground rod length [m] : %5.0f    ",
dat.rod_len);
} while (!item_selected);

```

```

    if (pos == 0)
        dat.D = tmp;
    else
        dat.rod_len = tmp;
    putimage(x0, y0, buf, COPY_PUT);
    farfree(buf);
    save_job_data(ff, dat);
}

void modify_job(dat_t &dat, tab_t *tab, char *ff)
{
    void far *buf1, far *buf2;
    char ch;
    double r, i, tmp;
    int x0, y0, x1, y1;
    int xx0 = 13, yy0 = 4, xx1 = 68, yy1 = 27;

    if (!strcmp(ff, NO_SELECT)) {
        no_job_error();
        return;
    }
    x0 = x2px(xx0);
    y0 = y2py(yy0);
    x1 = x2px(xx1);
    y1 = y2py(yy1);
    buf1 = farmalloc(imagesize(x0, y0, x1, y2py(yy0 + 9)));
    buf2 = farmalloc(imagesize(x0, y2py(yy0 + 9), x1, y1));
    getimage(x0, y0, x1, y2py(yy0 + 9), buf1);
    getimage(x0, y2py(yy0 + 9), x1, y1, buf2);
    draw_3d_plate(x0, y0, x1, y1);
    hsetbkcolor(DARKGRAY);
    hsetcolor(YELLOW);
    hgotoxy(xx0 + 2, yy0 + 1);
    hprintf("A) Job Name [87a ¶íÊü] : %-8s", dat.jobid);
    hgotoxy(xx0 + 2, yy0 + 2);
    hprintf("B) Required ground resist. [ohm] : %5.1f", dat.r_rg);
    hgotoxy(xx0 + 2, yy0 + 3);
    hprintf("C) Duration of shock [sec] : %5.1f", dat.ts);
    hgotoxy(xx0 + 2, yy0 + 4);
    hprintf("D) Fault duration [sec] : %5.1f", dat.tf);
    hgotoxy(xx0 + 2, yy0 + 5);
    hprintf("E) Fault impendace 1 [ohm] : %5.1f +j %5.1f",
        real(dat.z1), imag(dat.z1));
    hgotoxy(xx0 + 2, yy0 + 6);
    hprintf("F) Fault impedance 0 [ohm] : %5.1f +j %5.1f",

```

```

    }
    ch = toupper(ch);
    while (ch < 'B' || ch > 'S');
    if (ch == 'E' || ch == 'F') {
        r = get_data(xx0 + 37, yy0 + 5 + ch - 'E', "#999.9");
        i = get_data(xx0 + 45, yy0 + 5 + ch - 'E', "#999.9");
        if (ch == 'E')
            dat.z1 = complex(r, i);
        else
            dat.z0 = complex(r, i);
    }
    else if (ch == 'N') {
        dat.Ax = get_data(xx0 + 37, yy0 + 14, "#99999");
        dat.Ay = get_data(xx0 + 45, yy0 + 14, "#99999");
    }
    else if (ch == 'H') {
        dat.ln2ln_voltage = get_data(xx0 + 37, yy0 + 8, "#9999999");
    }
    else if (ch == 'G') {
        dat.i_f = get_data(xx0 + 37, yy0 + 7, "#9999999");
    }
    else {
        if (ch != 'Q')
            tmp = get_data(xx0 + 37, yy0 + 10 + ch - 'J',
"#9999.9");
        switch (ch) {
            case 'B':
                dat.r_rg = tmp;
                break;
            case 'C':
                dat.ts = tmp;
                break;
            case 'D':
                dat.tf = tmp;
                break;
            case 'I':
                dat.Sf = tmp;
                break;
            case 'J':
                dat.hs = tmp;
                break;
            case 'K':
                dat.rho_s = tmp;
                break;
            case 'L':

```

```

        dat.rhol = tmp;
        break;
    case 'M':
        dat.h = tmp;
        break;
    case 'O':
        dat.Tm = tmp;
        break;
    case 'P':
        dat.Ta = tmp;
        break;
    case 'Q':
        dat.material = select_material(tab, dat);
        hgotoxy(xx0 + 2, yy0 + 17);
        hprintf("Q) Grounding material          :
%17s", tab[dat.material].descript);
        break;
    case 'R':
        dat.D = tmp;
        break;
    case 'S':
        dat.rod_len = tmp;
        break;
    default:
        break;
    }
}
hsetcolor(LIGHTCYAN);
hgotoxy(xx0 + 2, yy1 - 2);
hputs("수정한 입력값이 정확합니까 [Y/N] ? ");
do {
    ch = toupper(getch());
} while (ch != 'Y' && ch != 'N');
if (ch == 'Y')
    save_job_data(ff, dat);
putimage(x0, y0, buf1, COPY_PUT);
putimage(x0, y2py(yy0 + 9), buf2, COPY_PUT);
farfree(buf1);
farfree(buf2);
save_job_data(dat, ff);
}
void print_result(dat_t dat, tab_t *tab, char *ff)
{
    FILE    *f;
    char    fd[13];

```

```

strcpy(fd, ff);
strcpy(&fd[9], "out");
f = fopen(fd, "wt");
fprintf(f, "(A) JOB id : %8s\n", dat.jobid);
fprintf(f, "(B) Conductor material : %15s\n",
tab[dat.material].descript);
fprintf(f, "(C) Duration of shock : %7.1f [sec]\n", dat.ts);
fprintf(f, "(D) Fault duration : %7.1f [sec]\n", dat.tf);
fprintf(f, "(E) Division factor : %7.1f\n", dat.Sf);
fprintf(f, "(F) Line to line voltage : %7.1f [kV]",
dat.ln2ln_voltage / 1000);
fprintf(f, "(G) 1st layer resistivity of soil : %7.1f [Ω-m]", dat.rho1);
fprintf(f, "(H) 2nd layer resistivity of soil : %7.1f [Ω-m]", dat.rho2);
fprintf(f, "(I) Resisitivity of cracks : %7.1f [Ω-m]", dat.rho_s);
fprintf(f, "(J) Depth of cracks : %7.1f [m]", dat.hs);
fprintf(f, "(K) Depth of grid : %7.1f [m]", dat.h);
fprintf(f, "(L) Available grounding area : %7.1f [m²]", dat.A);
fprintf(f, "(M) Maximum temperature : %7.1f [°C]",
dat.Tm);
fprintf(f, "(N) Ambient temperature : %7.1f [°C]", dat.Ta);
fprintf(f, "(O) Fault current : %7.1f [kA]", dat.i_f /
1000);
fprintf(f, "(P) Sym. ground fault current : %7.1f [A]", dat.i_g);
fprintf(f, "(Q) Ground resistance : %7.1f [Ω]", dat.Rg);
fprintf(f, "(R) Ground Potential Rise : %7.1f [V]", dat.i_g *
dat.Rg);
fprintf(f, "(S) Mesh voltage : %7.1f [V]", dat.Em);
fprintf(f, "(T) Reflection factor : %7.1f [V]", dat.K);
// fprintf(f, "(U) Deflection factor

fclose(f);
}

void main(void)
{
pd_menu menu("menu.dat");
int end, id;
char fjob[13];
tab_t tab[TAB_NO];
dat_t dat;

strcpy(fjob, NO_SELECT);
read_table_data("table.dat", tab);

```

```

userinithan();
logo();
clearviewport();
draw_3d_plate(x2px(28), y2py(25), x2px(51), y2py(28));
hgotoxy(29, 26);
hsetbkcolor(DARKGRAY);
hsetcolor(LIGHTCYAN);
hputs("JOB ID: Not Selected");
end = menu.select_finished(); // End-selection menu id
while ((id = menu.select_menu(0, menu.init_x(), menu.init_y())) !=
end) {
    switch (id) {
        case 111:
            create_new_job(dat, tab, fjob);
            if (strcmp(fjob, NO_SELECT)) {
                hgotoxy(37, 26);
                hputs("        ");
                hgotoxy(37, 26);
                hputs(fjob);
            }
            break;
        case 112:
            load_job(dat, tab, fjob);
            if (strcmp(fjob, NO_SELECT)) {
                hsetbkcolor(DARKGRAY);
                hsetcolor(LIGHTCYAN);
                hgotoxy(37, 26);
                hputs("        ");
                hgotoxy(37, 26);
                hputs(fjob);
            }
            break;
        case 113:
            view_job(dat, tab, fjob);
            break;
        case 121:
            modify_initial_design(fjob, dat);
            break;
        /*
        case 122:
            if (!strcmp(fjob, NO_SELECT))
                no_job_error();
            else {
                dat.material=select_material(tab, dat);
                save_job_data(fjob, dat);
            }
        */
    }
}

```

```

        break:*/
case 122:
    modify_job(dat, tab, fjob);
    break;
case 131:
    new_measurement_data(fjob);
    break;
case 133:
    calculate_rho(dat, fjob);
    break;
case 14:
    grounding_system_design(dat, tab, fjob);
    show_result(dat);
    save_job_data(fjob, dat);
    break;
case 151:
    show_result(dat);
    break;
case 152:
    break;
case 153:
    view_3d_graph("a.dat");
    break;
default:
    break;
    }
}
closehan();
cout << "\nGROUNDING SYSTEM DESIGN (C) 1994 KISCO,
Version 1.0\n";
}

```

산업용 플랜트 접지시스템의 전위분포에 관한 연구 (기전연 94-5-7)

발 행 일 : 1994. 12. 31

발 행 인 : 산업안전연구원장 서 상학

연구수행자 : 선임연구원 이 형수

발 행 처 : 한국산업안전공단

산업안전연구원

기계전기연구실

주 소 : 인천직할시 북구 구산동 34-4

전 화 : (032)513-0230

(032)502-0031~2

<비매품>