

중 간 보 고 서

- 조선해양공학계획 -

3조 데시그나레(Designare)

2002-12047 손지호

2005-11931 김영민

2005-11962 박한솔

2005-11984 양해상

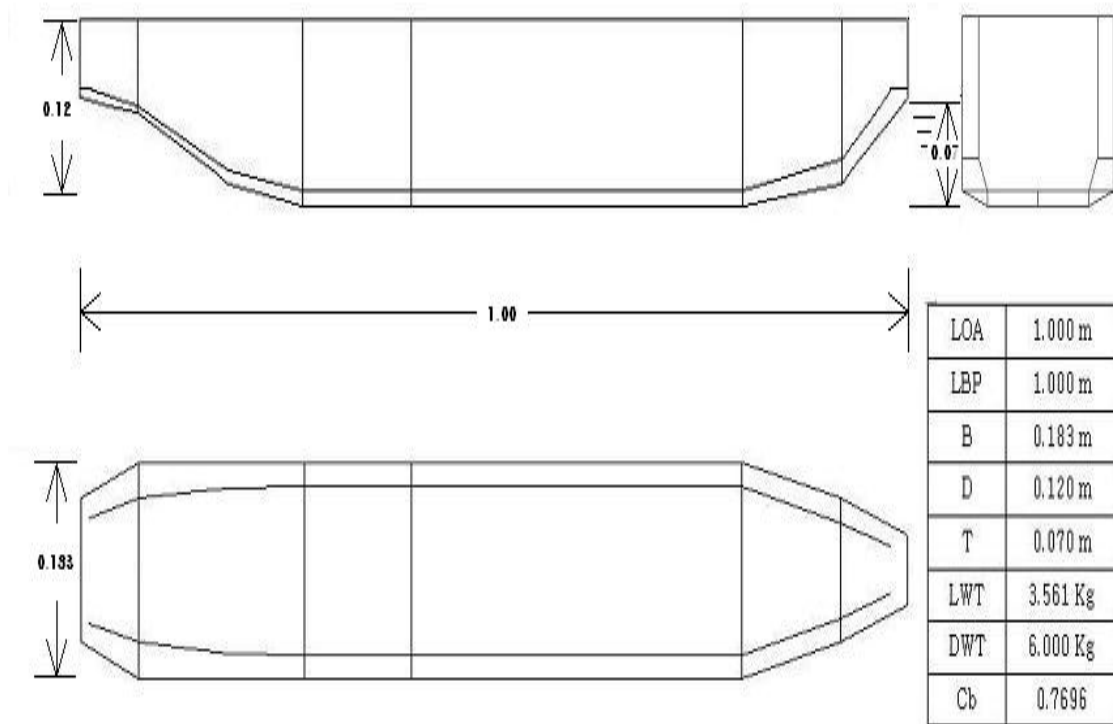
2005-11991 윤상훈

2005-12031 주경환

요 약

1. 주요요목

* GA



* LWT(상세히)

선체부	2.56084277 kg
기관부	1 kg
합계	3.56084277 kg

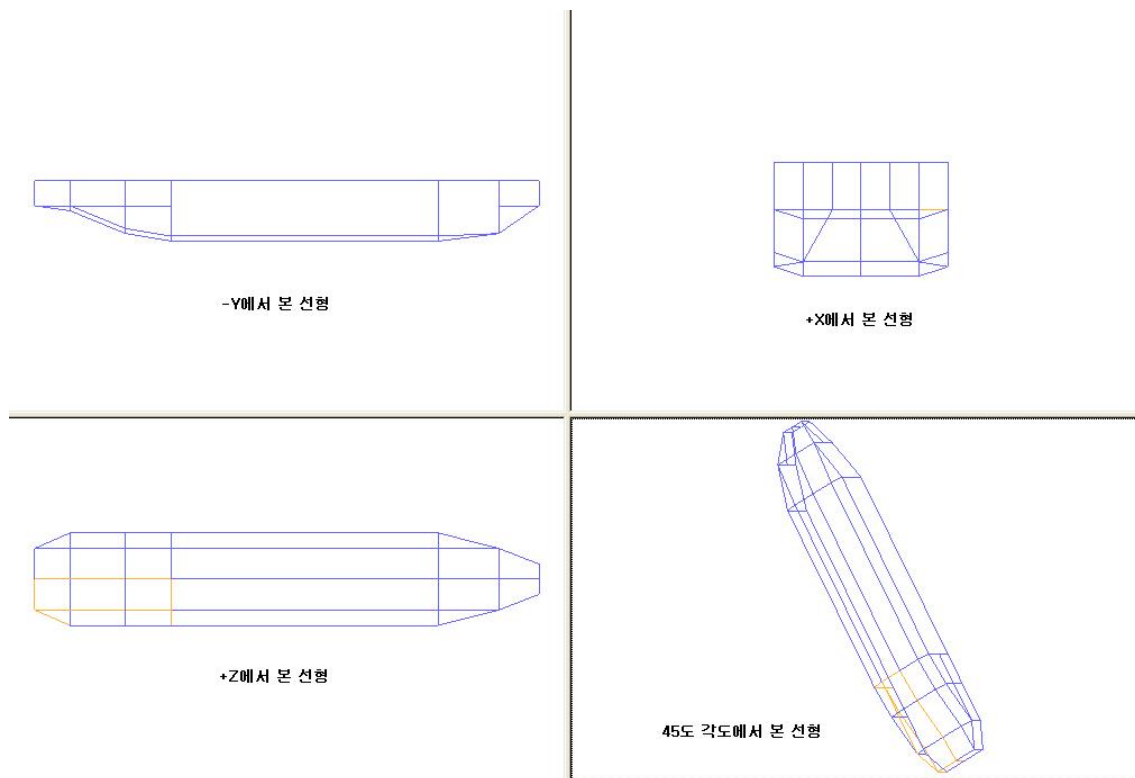
* 배수량

배수량 = 방형계수 × LBP × Breadth × Draft × Water's Density

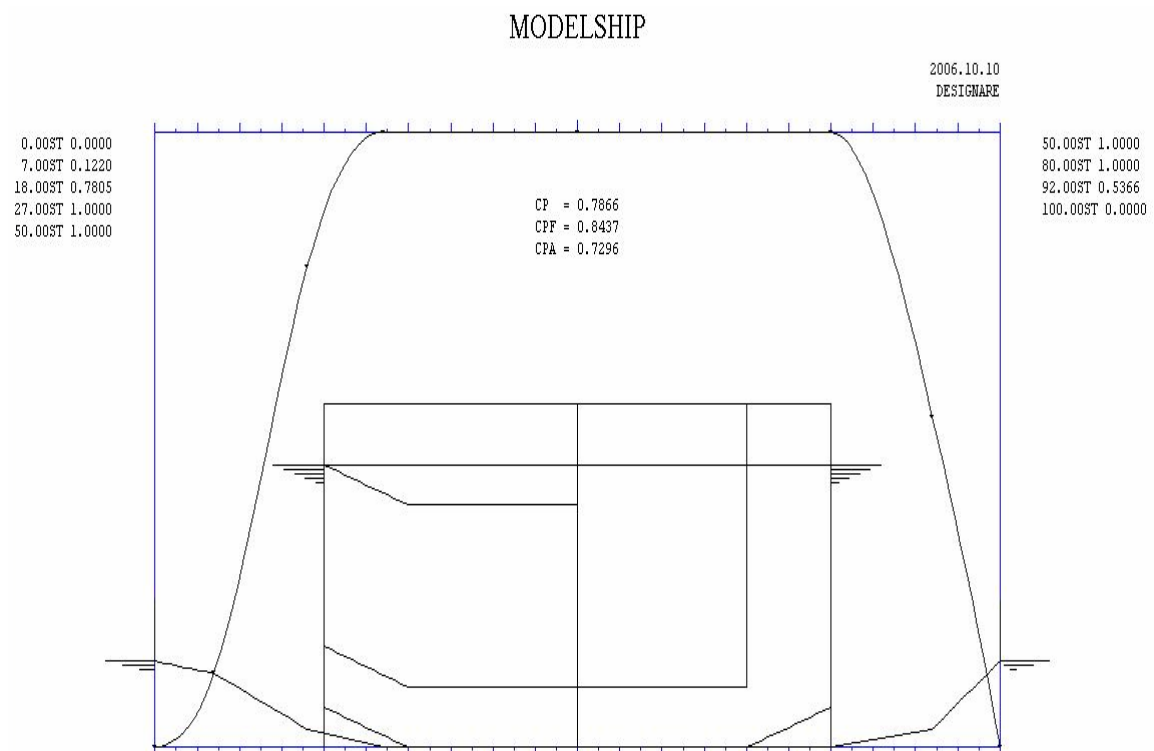
이므로

$$0.7917 \times 0.95 \times 0.183 \times 0.06 \times 10^6 \times 1 = 8258.223 \text{ g}$$

2. 선형



3. Cp 커브



L O A = 1.00 M
L B P = 1.00 M
BEAM = 0.18 M
DEPTH = 0.12 M
DRAFT = 0.07 M(TF)
= 0.07 M(TA)
SPEED = 0.00 KTS
FN = 0.0000
SCH = 0.000 M
DIA = 0.000 M

L/B = 5.4645
B/T = 2.6143
CB = 0.7679
CBF = 0.8236
CBA = 0.7122
CM = 0.9762
CW = 0.9341
CWF = 0.8915
CWA = 0.9767
CB/(L/B) = 0.1405

LCB = 2.83 % LPP
VCB = 0.04 M
KMT = 0.0822 M
LE = 0.200 M(20.00 %)
LX = 0.530 M(53.00 %)
LR = 0.270 M(27.00 %)
AFP = 0.000 M^2(0.00 % AM)
BL = 0.000 M(0.00 % LPP)
WSA = 0.3 M^2
VOL = 0.0 M^3

목 차

1. 배의 설계.....	5
1.1. 사용틀 -----	5
1.2. 우리들이 예상했던 치수와 GA -----	5
1.3. 선형 설계 -----	6
1.3.1. 1차설계 -----	6
1.3.2. 2차설계 -----	8
1.3.3. 3차설계 -----	10
1.4. 구획배치 -----	13
1.4.1. 구획배치의 모양 -----	13
1.4.2. 구획배치의 평가 -----	13
1.5. 경하중량의 추정 -----	14
1.5.1. 배수량 -----	14
1.5.2. 경하중량 추정을 위한 구획배치도 -----	14
1.5.3. 선미, 선수, 중앙 평행부 부분 프레임과 격벽의 넓이 -----	15
1.5.4. 선미, 선수, 중앙 평행부 부분의 Deck의 넓이 -----	16
1.5.5. Z-축 프레임의 넓이 -----	16
1.5.6. 용골의 넓이 -----	16
1.5.7. 경하중량의 계산 -----	17
1.6. 4차설계 -----	18
1.6.1. 선형설계 -----	19
1.6.2. 4차설계의 평가 -----	20
1.6.3. 구획배치 -----	21
1.7. 경하중량의 재추정 -----	22
1.7.1. 배수량 -----	22
1.7.2. 부재 넓이의 계산 -----	22
1.7.3. 경하중량의 재계산 -----	23
2. 제어부.....	25
2.1. 기본 소자 조사 -----	25
2.1.1. 마이크로프로세서(MPU)와 마이크로컨트롤러(MCU) -----	25
2.2. AVR 조사 -----	30

2.2.1.	AVR의 종류 -----	31
2.2.2.	AVR의 특징 -----	32
2.2.3.	AVR의 개발환경 -----	32
2.3.	ATMEGA 8535 조사 -----	33
2.3.1.	ATMEGA 8535의 특징 -----	33
2.3.2.	ATMEGA 8535의 구조 -----	34
2.3.3.	ATMEGA 8535의 메모리 구조 -----	36
2.4.	각 부분의 회로도. -----	38
2.4.1.	Rom Writer Cable의 회로도 -----	38
2.4.2.	cpu 보드의 회로도 -----	39
2.4.3.	Rom Writer + cpu보드의 회로도 -----	40
2.4.4.	서보모터와 DC모터의 회로도 -----	40
2.4.5.	우리 팀 프로젝트와 관련하여 주어진 과제 (LED 점등 회로도) ---	42
2.4.6.	센서연결을 위한 AVR 회로도 -----	42
2.5.	원격 조정부 (수동 조정부) -----	43
2.5.1.	기본 개념 조사 -----	43
2.6.	크리스탈 -----	46
2.6.1.	기본개념 조사 -----	46
2.7.	추진부 -----	47
2.7.1.	서보모터 [servomotor]-----	47
2.7.2.	전동기 [電動機, electric motor] -----	48
2.7.3.	필요한 개념 -----	49
3.	세부 제작 일정	51
4.	역할 분담.....	52
5.	전체 후기.....	52

그 립 목 차

그림 1.1. 예상했던 치수와 GA -----	5
그림 1.2. 1차 설계의 단면도 -----	6
그림 1.3. 1차 설계의 측면도 -----	6
그림 1.4. 1차 설계의 평면도 -----	6
그림 1.5. 1차 설계의 Cp 커브. -----	7
그림 1.6. 2차 설계의 단면도 -----	8
그림 1.7. 2차 설계의 측면도 -----	8
그림 1.8. 2차 설계의 평면도 -----	8
그림 1.9. 2차 설계의 Cp 커브 -----	9
그림 1.10. 3차 설계의 단면도 -----	10
그림 1.11. 3차 설계의 측면도 -----	10
그림 1.12. 3차 설계의 평면도 -----	10
그림 1.13. 3차 설계의 Cp 커브 -----	11
그림 1.14. 최종선형설계 모형 -----	12
그림 1.15. 구획의 단면도 -----	13
그림 1.16. 구획의 측면도 -----	13
그림 1.17. 구획의 평면도 -----	13
그림 1.18. 단면도 -----	14
그림 1.19. 측면도 -----	14
그림 1.20. 평면도 -----	14
그림 1.21. 프레임의 넓이 비 -----	15
그림 1.22. 용골 -----	17
그림 1.23. 단면도 -----	19
그림 1.24. 측면도 -----	19
그림 1.25. 평면도 -----	19
그림 1.26. Cp 커브 -----	20
그림 1.27. 구획배치 단면도 -----	21
그림 1.28. 구획배치 측면도 -----	21
그림 1.29. 구획배치 정면도 -----	21
그림 2.1. 마이크로프로세서 -----	25
그림 2.2. AVR -----	27
그림 2.3. Atmega8535 -----	34
그림 2.4. ATMEGA8535의 메모리 구조 -----	36

그림 2.5. MCU와 PC간의 정보교환 모식도 -----	37
그림 2.6. Rom Writer Cable의 회로도-----	38
그림 2.7. CPU 보드의 회로도 -----	39
그림 2.8. Rom Wrinter + CPU 보드의 회로도 -----	40
그림 2.9. 서보모터와 DC모터의 회로도-----	40
그림 2.10. Parallel port 핀-----	41
그림 2.11. LED 점등 회로도-----	42
그림 2.12. 센서연결을 위한 AVR 회로도 -----	42
그림 2.13. DSB AM 파의 변조과정 -----	43
그림 2.14. DSB-SC AM 파의 변조파형 -----	44
그림 2.15. AM의 변조지수 -----	44
그림 2.16. FM 변조의 원리 -----	45
그림 2.17. 변조지수에 따른 FM의 주파수분포-----	45
그림 2.18. 직류전동기의 회전원리 -----	48
그림 2.19. 로렌츠의 힘 -----	50

1. 배의 설계

1.1. 사용툴

EZship을 이용하여 배를 설계하도록 한다.

1.2. 우리들이 예상했던 치수와 GA

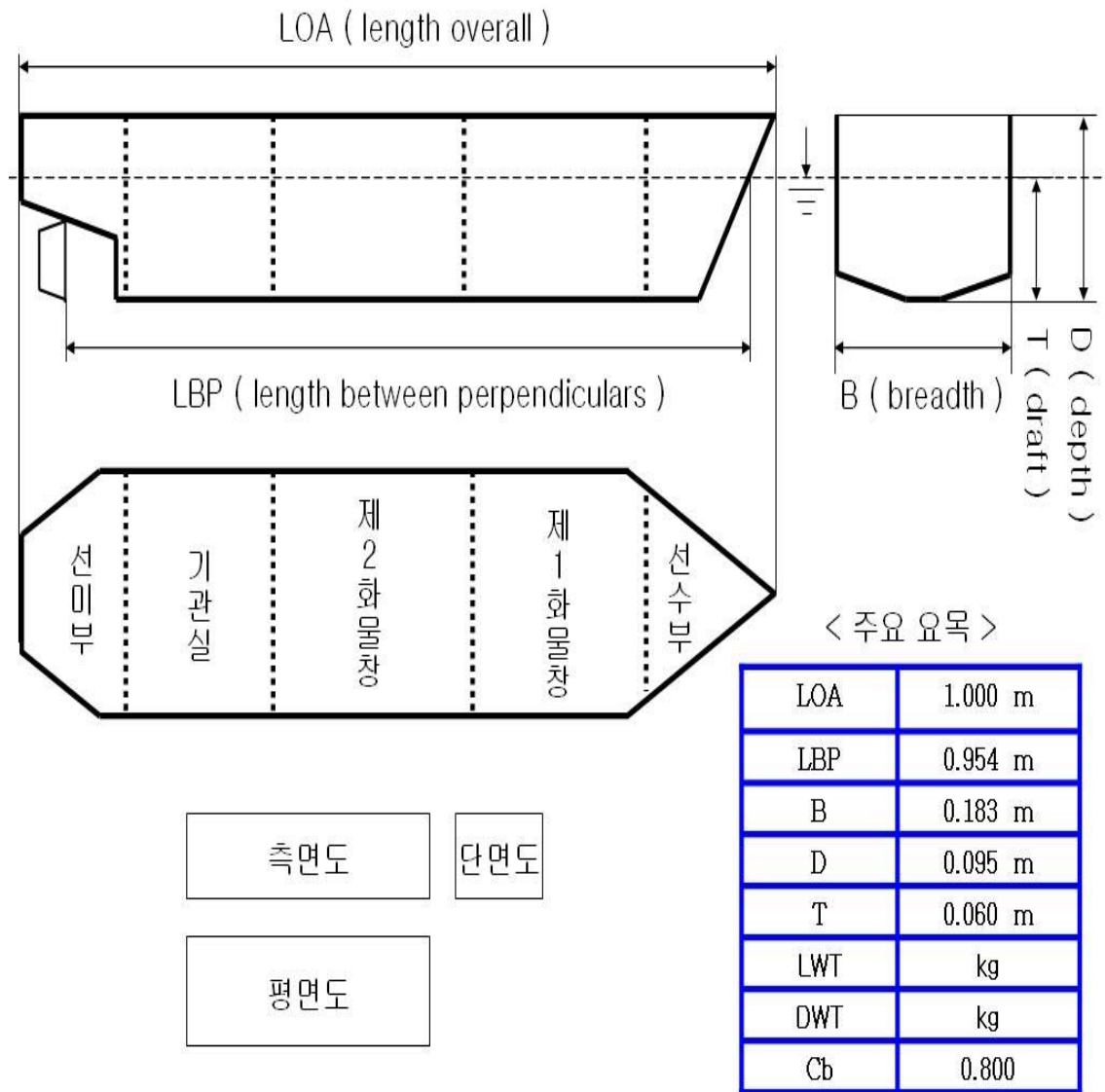


그림 1.1. 예상했던 치수와 GA

1.3. 선형 설계

1.3.1. 1차설계

1.3.1.1. 1차설계 모양

* 단면도

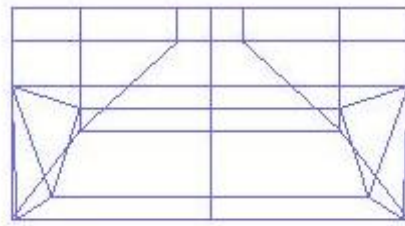


그림 1.2. 1차 설계의 단면도

* 측면도



그림 1.3. 1차 설계의 측면도

* 평면도



그림 1.4. 1차 설계의 평면도

* Cp 커브

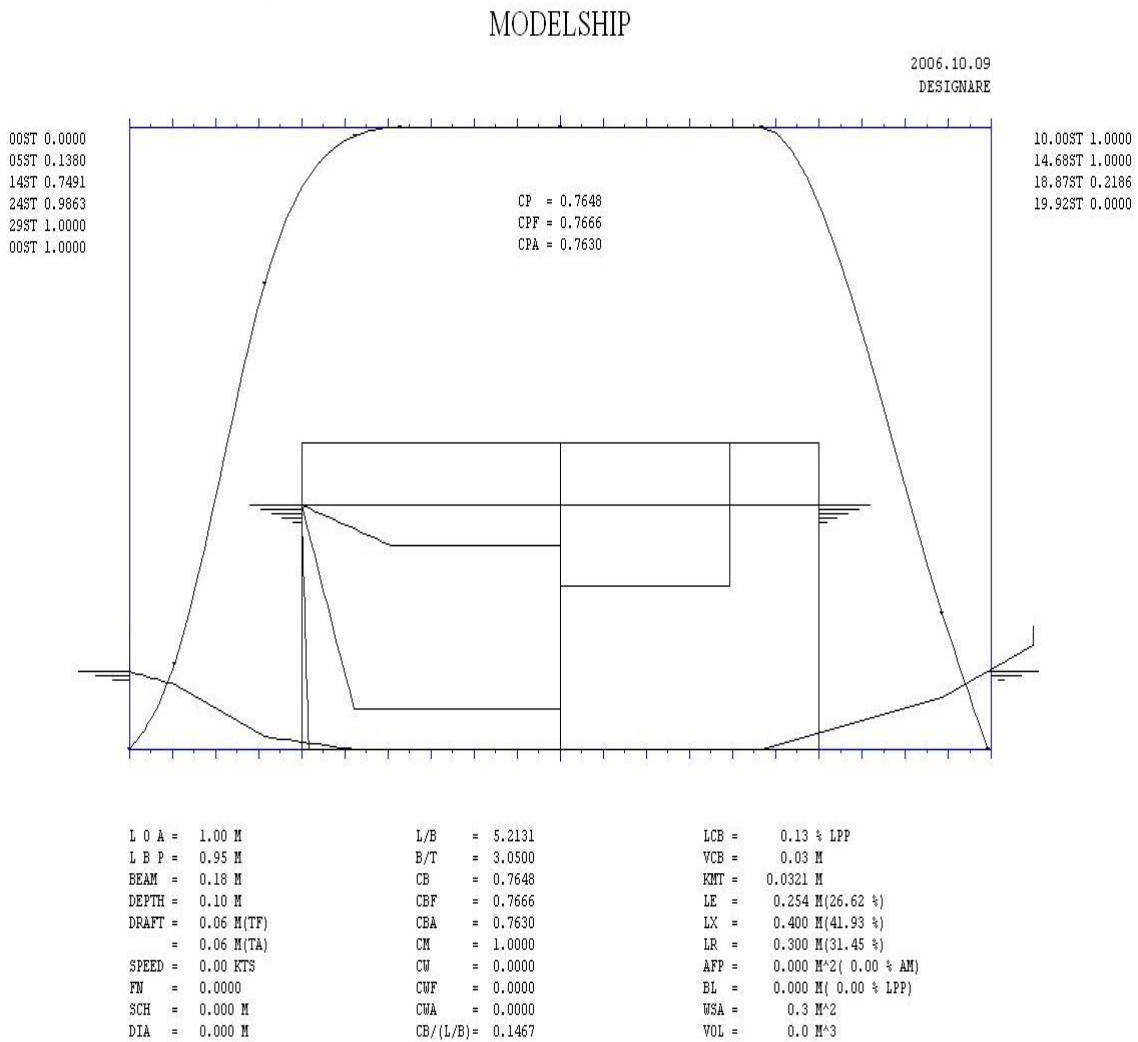


그림 1.5. 1차 설계의 Cp 커브.

1.3.1.2. 1차설계의 평가

보다시피 첫 설계의 Cp 커브가 원만한 곡선을 이루고 있어서 적합한 선형을 이루었다고 판단했다. 하지만 우리가 예상했던 방형계수가 0.8이었는데 이번 설계에서는 0.7648이 나와서 기대치보다 좀 작으므로 흘수 아래의 부피를 늘리기 위해 약간의 수정을 거치기로 하였다.

1.3.2. 2차설계

1.3.2.1. 2차설계 모양

* 단면도

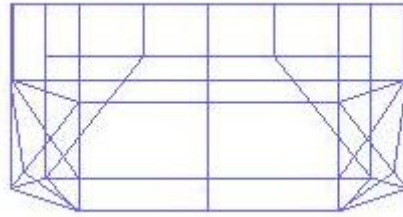


그림 1.6. 2차 설계의 단면도

* 측면도



그림 1.7. 2차 설계의 측면도

* 평면도



그림 1.8. 2차 설계의 평면도

* C_p 커브

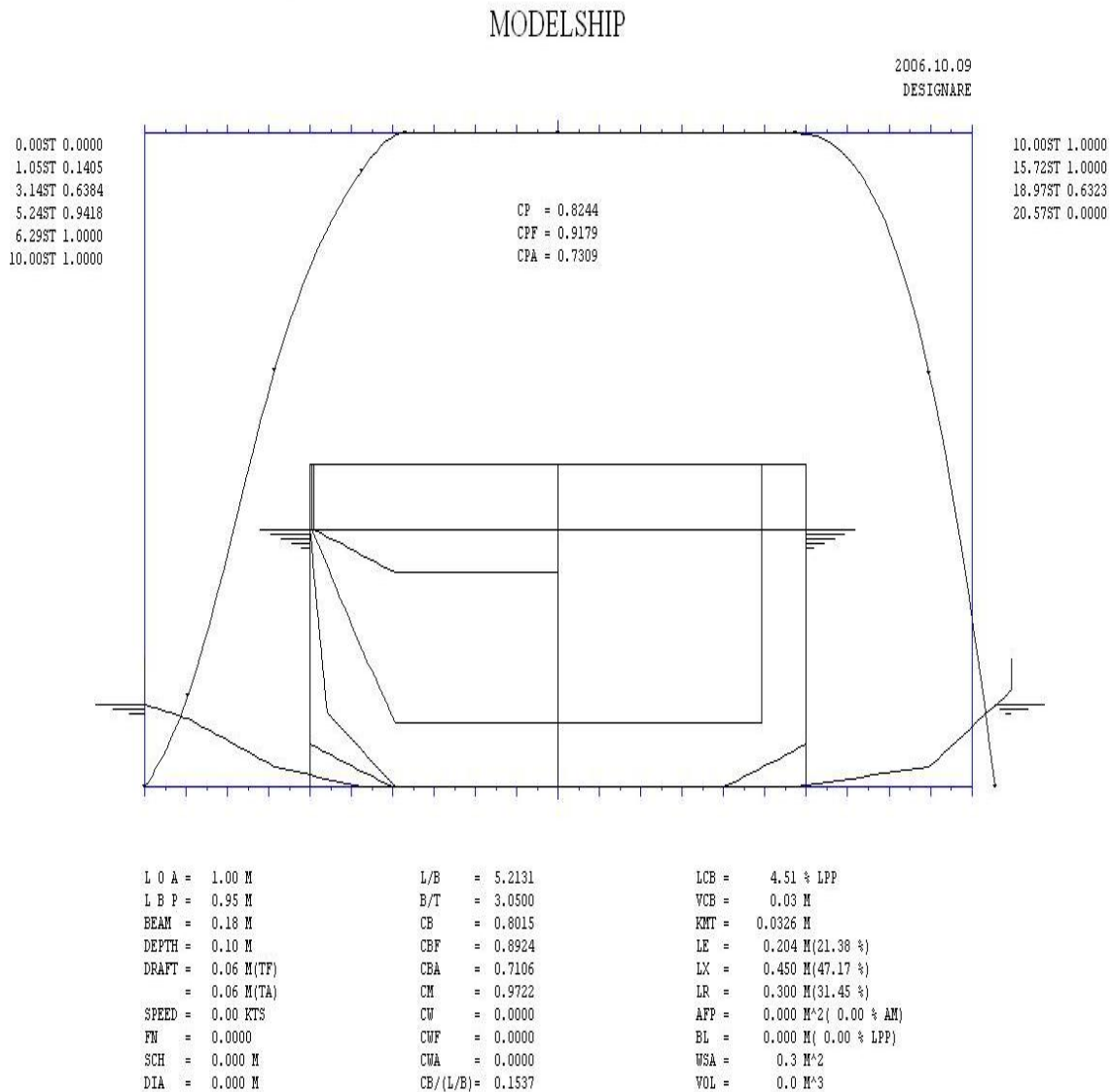


그림 1.9. 2차 설계의 C_p 커브

1.3.2.2. 2차설계의 평가

부피를 조금 늘린 결과 일단 C_p 커브는 이번에도 무난한 곡선을 나타내어 선형에는 별 문제가 없음을 드러냈고, 방형계수 역시 상승하여 0.8015의 값을 나타내었다. 하지만 이번 설계에서는 건현이 너무 낮다는 안전 상의 문제가 있었다. 그래서 높이를 좀 높이는 작업을 다시 하여 수정하기로 결정했다.

1.3.3. 3차설계

1.3.3.1. 3차설계 모양

* 단면도

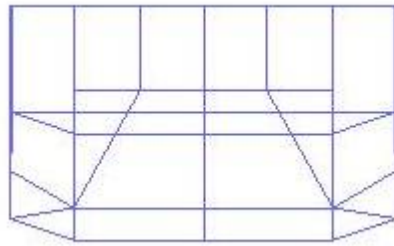


그림 1.10. 3차 설계의 단면도

* 측면도



그림 1.11. 3차 설계의 측면도

* 평면도

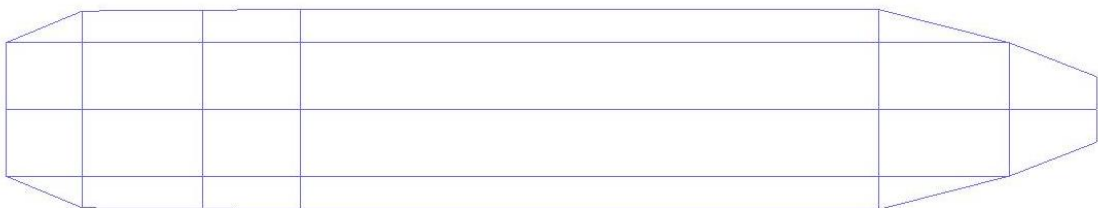


그림 1.12. 3차 설계의 평면도

* Cp 커브

MODELSHIP

2006.10.09
DESIGNARE

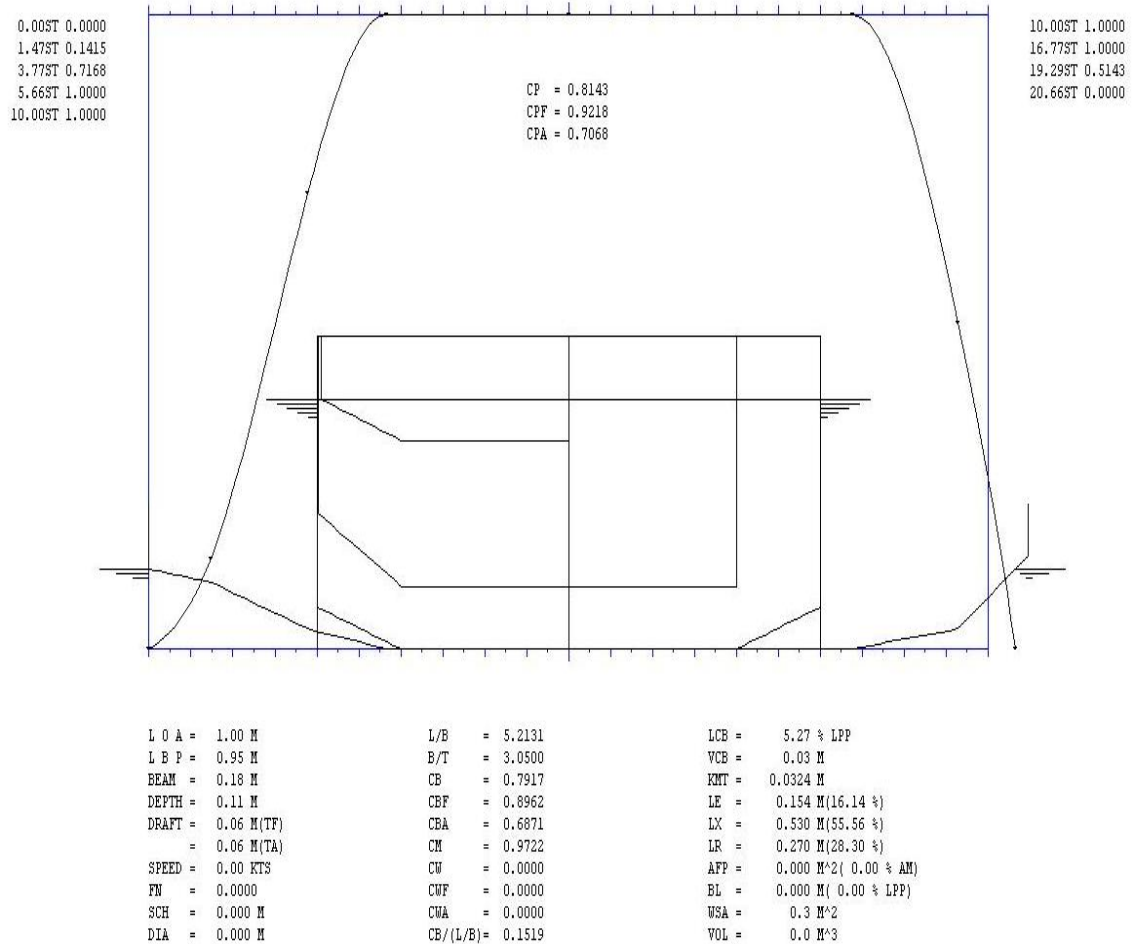


그림 1.13. 3차 설계의 C_p 커브

1.3.3.2. 3차설계의 평가

이번 3번째 설계에서는 건현의 높이를 좀 높이고, 부피를 약간 줄이는 작업을 했다. 그 결과 여전히 Cp 커브는 원만한 곡선을 나타내어 선형에는 아무런 이상이 없었고 건현의 높이도 안전상의 이상이 없어 이번에 설계한 선형으로 구획배치로 넘어가기로 했다.

1.3.3.3. 최종선형

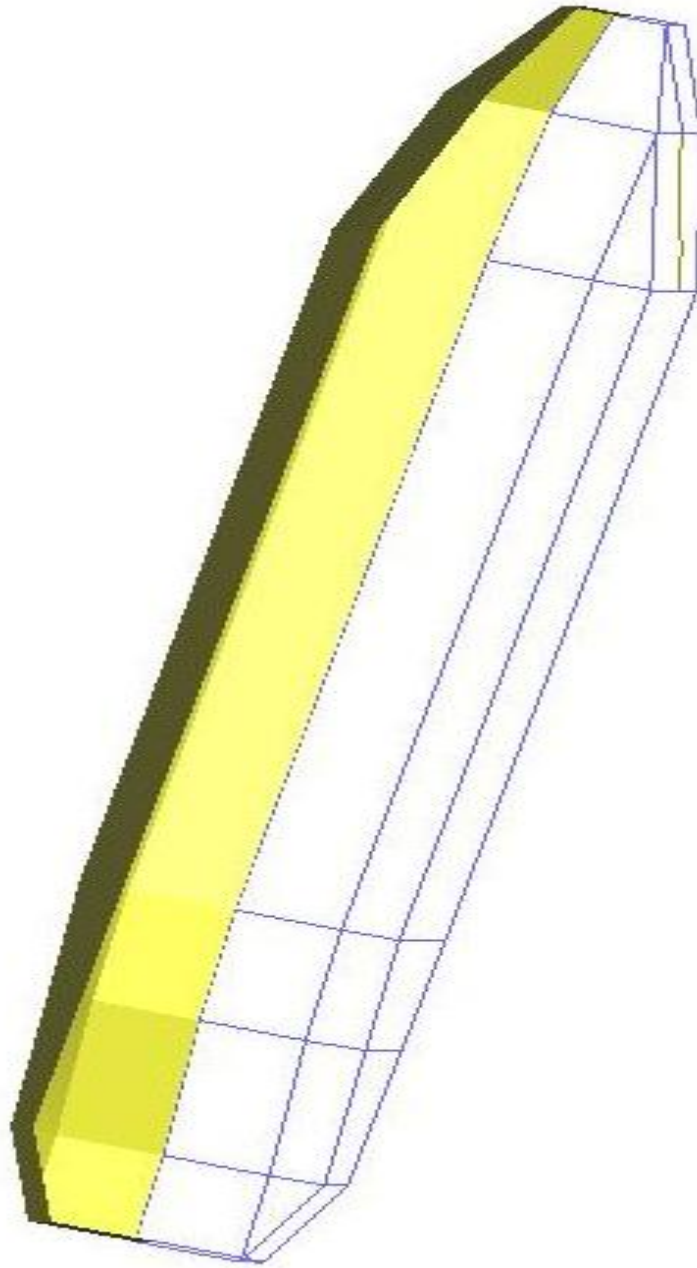


그림 1.14. 최종선형설계 모형

1.4. 구획배치

1.4.1. 구획배치의 모양

*구획의 단면도

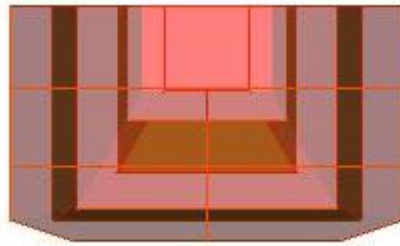


그림 1.15. 구획의 단면도

* 구획의 측면도

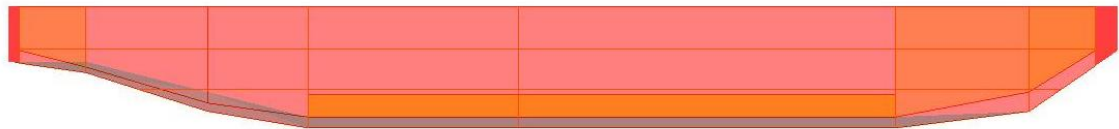


그림 1.16. 구획의 측면도

* 구획의 평면도

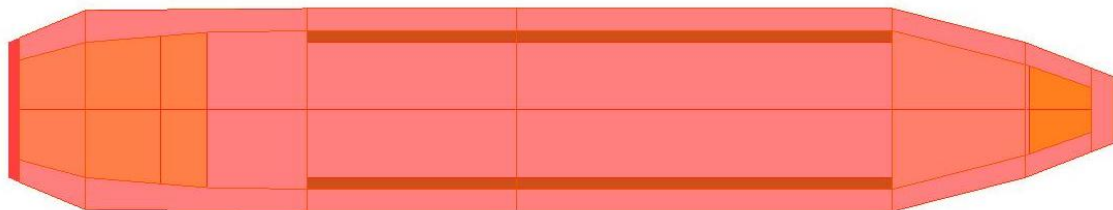


그림 1.17. 구획의 평면도

1.4.2. 구획배치의 평가

처음에 예상했던 대로 격벽을 4개로 나누려 하였으나 선수부분에 센서를 3개를 장착할 것을 예상하고 있으므로 이를 구분하기 위하여 선수부에 격벽을 하나 더 넣어 5개로 하기로 수정을 하였다.

1.5. 경하중량의 추정

1.5.1. 배수량

경하중량을 추정하기에 앞서 배수량을 계산하여 보면,

배수량 = 방형계수 × LBP × Breadth × Draft × Water's Density

이므로

$$0.7917 \times 0.95 \times 0.183 \times 0.06 \times 10^6 \times 1 = 8258.223 \text{ g}$$

이 나온다.

1.5.2. 경하중량 추정을 위한 구획배치도

경하중량을 추정하기 위해 구획을 좀더 세분히 나눠보았다.

*단면도

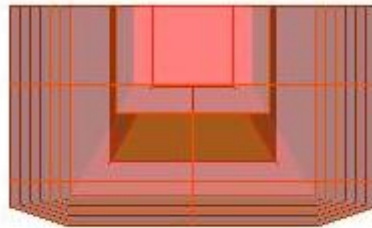


그림 1.18. 단면도

*측면도

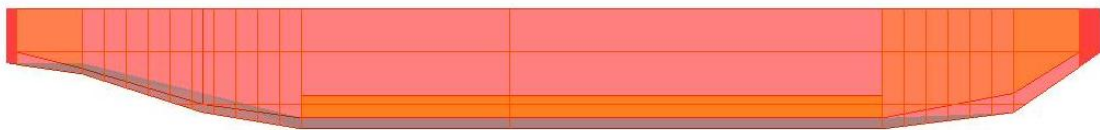


그림 1.19. 측면도

*평면도

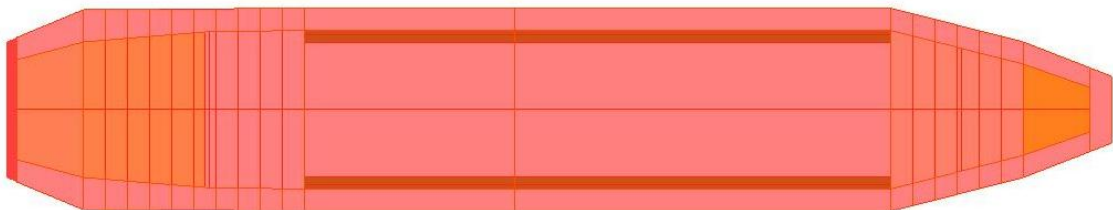
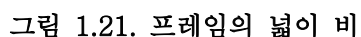


그림 1.20. 평면도

1.5.3. 선미, 선수, 중앙 평행부 부분 프레임과 격벽의 넓이


$$\text{선미} : \frac{0.005525 \times (1 - (0.95)^{20})}{1 - 0.95^2} - 0.005525 = 0.030827464397 \text{ } m^2$$
$$\text{전수} : \frac{0.005525 \times (1 - (0.99167)^{12})}{1 - 0.99167^2} - 0.005525 = 0.026280092103 \text{ } m^2$$

평행부 부분의 프레임은 가장 큰 프레임과 같은 크기가 25개 있는 것 이므로

$$0.005525 \times 25 = 0.138125 \text{ m}^2$$

가장 큰 프레임에서 가운데 네모난 직사각형을 더하면 가장 큰 격벽과 같은 크기가 되므로 구하여 보면 0.019825 m^2 이 된다.

격벽중에서도 작은 격벽은 넓이 비를 이용해 구하면 $0.019825 \times 0.95^{20} \text{ m}^2$ 과 $0.019825 \times 0.99167^{12} \text{ m}^2$ 이다. 가장 큰 격벽은 3장이므로 격벽의 총 넓이는 $0.019825 \times 3 + 0.019825 \times 0.95^{20} + 0.019825 \times 0.99167^{12} = 0.084513593955 \text{ m}^2$ 이다.

1.5.4. 선미, 선수, 중앙 평행부 부분의 Deck의 넓이

우리 배의 Deck는 사다리꼴과 직사각형으로 이루어져 있고, 좌우 대칭을 이루고 있다. 따라서 배 반쪽의 넓이를

$$\text{사다리꼴의 넓이} = \frac{(\text{밑변} + \text{윗변}) \times \text{높이}}{2}$$

직사각형의 넓이 = 밑변 \times 높이

를 이용하여 구하기로 했다.

변의 길이는

$$\text{점과 점 사이의 거리} = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

을 이용하여 구했으며 총 구한 넓이는

선미 : $0.07294268084 \text{ m}^2$

선수 : $0.047497873482 \text{ m}^2$

중앙 평행부 : $0.173404036836 \text{ m}^2$

이 나왔다.

1.5.5. Z-축 프레임의 넓이

Z-축 프레임 또한 직사각형과 사다리꼴로 이루어져있다. 앞서 말한 식들을 이용하여 구하되, 위와 아래 총 두개가 있다.

위쪽 프레임 넓이 = 0.04248 m^2

아래쪽 프레임 넓이 = $0.07030315899 \text{ m}^2$

따라서 Z-축 프레임의 총 넓이는

위쪽 프레임 넓이 + 아래쪽 프레임 넓이 = $0.11278315899 \text{ m}^2$

이 된다.

1.5.6. 용골의 넓이

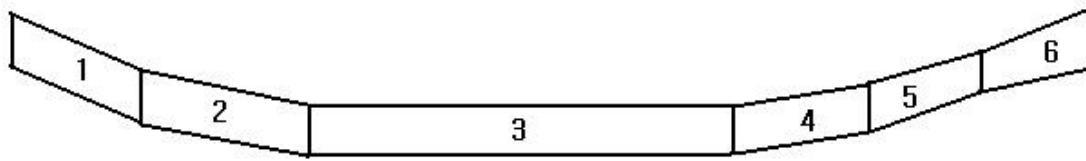


그림 1.22. 용골

용골은 위의 그림과 같이 6개의 부분으로 나누어서 계산하였다,

1번은 평행사변형에 매우 근사하여 평행사변형 넓이 구하듯이 구하였고, 3번은 직사각형이므로 세로와 가로를 곱하여 구하였다. 그리고 나머지 2,4,5,6번은 사다리꼴이므로 사다리꼴의 넓이를 구하는 공식을 써서 구하였다. 그 넓이는 다음과 같다.

$$1 = 0.00105$$

$$2 = 0.00165$$

$$3 = 0.053$$

$$4 = 0.0007875$$

$$5 = 0.00044$$

$$6 = 0.00049287$$

따라서 용골의 총 넓이는

$$1 + 2 + 3 + 4 + 5 + 6 = 0.00972037 \text{ } m^2$$

이다.

1.5.7. 경하중량의 계산

앞서 구한 것을 모두 살펴보면 (단위 : cm^2)

$$\text{격벽의 넓이} = 845.136$$

$$\text{용골의 넓이} = 97.2037$$

$$\text{프레임의 넓이} = 1976.907$$

$$\text{Deck의 넓이} = 2938.446$$

선형설계 후 C_p 커브에서

L O A =	1.00 M	L/B =	5.2131	LCB =	2.97 % LPP
L B P =	0.95 M	B/T =	1.6636	VCB =	0.06 M
BEAM =	0.18 M	CB =	0.8678	KMT =	0.0848 M
DEPTH =	0.11 M	CBF =	0.9295	LE =	0.154 M(16.14 %)
DRAFT =	0.11 M(TF)	CBA =	0.8061	LX =	0.530 M(55.56 %)
	0.11 M(TA)	CM =	0.9848	LR =	0.270 M(28.30 %)
SPEED =	0.00 KTS	CW =	0.9705	AFP =	0.000 M^2(0.00 % AM)
FN =	0.0000	CWF =	0.9702	BL =	0.000 M(0.00 % LPP)
SCH =	0.000 M	CWA =	0.9709	WSA =	0.4 M^2
DIA =	0.000 M	CB/(L/B) =	0.1665	VOL =	0.0 M^3

(홀수를 배의 높이와 같게 맞추어 표면적을 계산하였다. 빨간 밑줄 참고)

따라서

선체표면의 넓이 = 4000

수업자료로부터 모은 정보에서

선체의 두께 = 0.2cm

퍼티의 두께 = 0.05cm

에폭시의 두께 = 0.05cm

기관부의 무게 = 1 kg¹

우드락 밀도 = 0.7 g/cm³

퍼티의 밀도 = 1.35 g/cm³

에폭시의 밀도 = 1.5 g/cm³

Deck와 격벽과 용골, 선체의 두께는 0.2cm으로 하고 프레임의 두께를 0.1cm으로 하여 계산을 해 본 결과

$$4000 \times 0.2 \times 0.7 + 845.136 \times 0.2 \times 0.7 + 97.2037 \times 0.2 \times 0.7 + 1976.907 \times 0.1 \times 0.7 + 4000 \times 0.05 \times 1.35 + 4000 \times 0.05 \times 1.5 + 2938.446 \times 0.2 \times 0.7 + 820 = 2631.693488 \text{ g}$$

앞서 추정한 배수량이 8258.233g이었으므로 이 배가 실을 수 있는 재화중량은

$$\text{배수량} - \text{경화중량} = 5626.539512 \text{ g}$$

이다.

근데 이 재화중량은 이번에 만드는 배의 조건인 재화중량 6kg을 충족시키지 못한다.

결국 홀수를 높여서 배수량을 확보하기로 결정하였다. 하지만 홀수를 높이게 되면 상대적으로 건현의 길이가 짧아지게 되므로 선박의 안정성 확보를 위하여 선박의 깊이 또한 같이 바꾸는 것이 불가피하게 되었다. 그래서 기존의 치수는 그대로 유지하되 홀수와 깊이를 각각 1cm씩 늘리기로 결정하였다.

교정된 치수에 맞게 다시 선형작업과 구획배치작업을 하였다

1.6. 4차설계

¹ 수업자료에는 0.74kg로 되어있으나 빠진 부품이 많아 그것까지 계산하였다.

1.6.1. 선형설계

*단면도

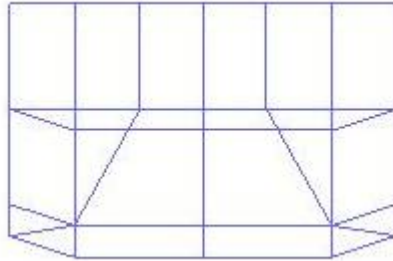


그림 1.23. 단면도

*측면도



그림 1.24. 측면도

*평면도

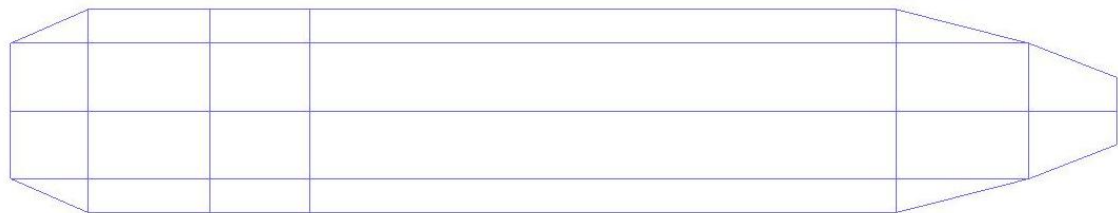


그림 1.25. 평면도

* Cp 커브

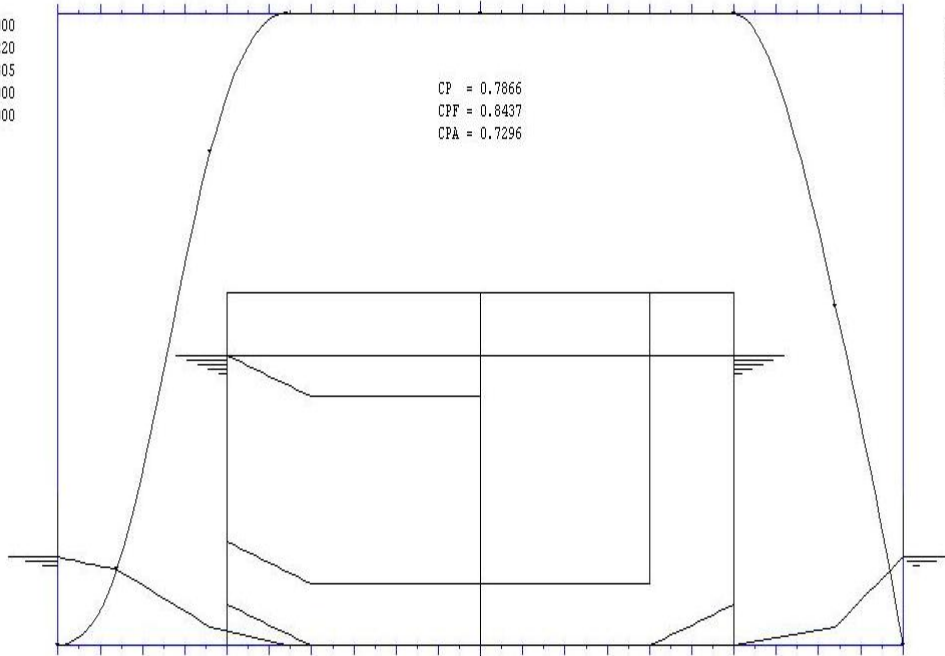
MODELSHIP

2006.10.10
DESIGNARE

0.00ST 0.0000
7.00ST 0.1220
18.00ST 0.7805
27.00ST 1.0000
50.00ST 1.0000

50.00ST 1.0000
80.00ST 1.0000
92.00ST 0.5366
100.00ST 0.0000

CP = 0.7866
CPF = 0.8437
CPA = 0.7296



LOA = 1.00 M
LBP = 1.00 M
BEAM = 0.18 M
DEPTH = 0.12 M
DRAFT = 0.07 M(TF)
= 0.07 M(TA)
SPEED = 0.00 KTS
FN = 0.0000
SCH = 0.000 M
DIA = 0.000 M

L/B = 5.4645
B/T = 2.6143
CB = 0.7679
CBF = 0.8236
CBA = 0.7122
CM = 0.9762
CW = 0.9341
CWF = 0.8915
CWA = 0.9767
CB/(L/B) = 0.1405

LCB = 2.83 % LPP
VCB = 0.04 M
KMT = 0.0822 M
LE = 0.200 M(20.00 %)
LX = 0.530 M(53.00 %)
LR = 0.270 M(27.00 %)
AFP = 0.000 M²(0.00 % AM)
BL = 0.000 M(0.00 % LPP)
WSA = 0.3 M²
VOL = 0.0 M³

그림 1.26. Cp 커브

1.6.2. 4차설계의 평가

3차 설계의 결과를 가지고 구획배치를 해 본 결과 우리가 예상했던 경화중량보다 큰 값이 나왔다. 이는 처음에 경화중량 추정을 잘못된 것도 있지만 주요 치수 결정에서 충분한 배수량을 확보하지 못한 결과이기에 4차 설계에서는 선박의 안정성과 충분한 배수량 확보를 중점으로 하였다. 4차 설계를 거친 결과 충분한 배수량을 확보할 수 있었다.(추후에 계산 하겠다.) 배수량이 늘어난 덕분에 용골을 2개 더 추가하는 등 더 많은 내부부재를 사용 할 수 있었다. 애초에 계획했던 방형계수를 얻지 못하였지만 매끄러운 Cp 커브 곡선을 얻을 수 있었고 용골을 추가하는 등 충분한 안전성을 확보하였다고 판단하여 방형계수를 0.7696으로 바꾸기로 결정하였다.

1.6.3. 구획배치

*단면도

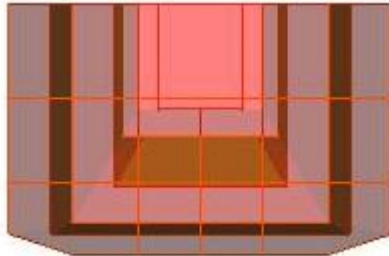


그림 1.27. 구획배치 단면도

*측면도

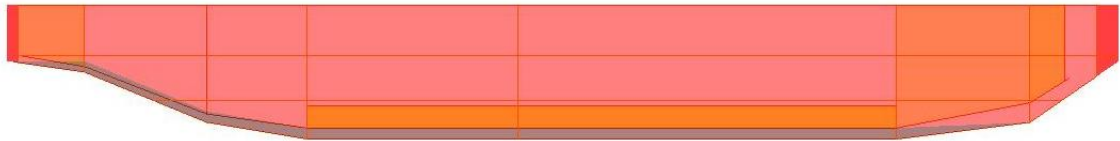


그림 1.28. 구획배치 측면도

*평면도

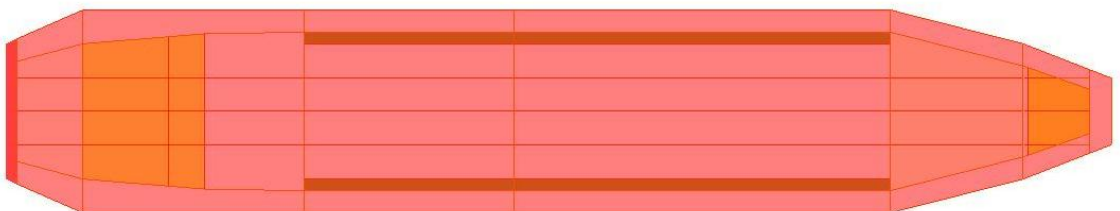


그림 1.29. 구획배치 정면도

1.7. 경하중량의 재추정

1.7.1. 배수량

이번에는 방형계수가 0.7696이므로 배수량은

$$0.7696 \times 0.183 \times 0.07 \times 10^6 \times 1 = 9858.576 \text{ g}$$

1.7.2. 부재 넓이의 계산

이번 부재의 넓이는 앞서 계산했던 부재의 넓이들을 기본으로 하여 계산하였다. 우선 Ezcompart 의 소스를 처음에 했던 것을 거의 그대로 가져다 썼을 정도로 선형은 거의 바뀌지 않았다. 이는 Cp 커브 곡선이 거의 변하지 않은 것만 봐도 알 수 있다. 다만 용골이 3개로 늘어났기 때문에 그 늘어난 넓이를 계산하고, 배의 높이가 위로 1cm씩 늘어났기 때문에 부재들의 늘어난 넓이는 직사각형일 것이므로 이 것에 근거하여 늘어난 넓이만을 계산하기로 결정하였다.

1.7.2.1. 선미, 선수, 중앙 평행부 부분 프레임과 격벽의 넓이

선미, 선수, 중앙 평행부, 격벽 모두 위에만 직사각형 형태로 늘어났다. 따라서 그 늘어난 부분의 넓이를 계산해 보면,

$$\text{선미의 늘어난 넓이} = 0.002231852626 \text{ m}^2$$

$$\text{중앙 평행부의 늘어난 넓이} = 0.01 \text{ m}^2$$

$$\text{선수의 늘어난 넓이} = 0.003669204865 \text{ m}^2$$

$$\text{격벽의 늘어난 넓이} = 0.007801254827 \text{ m}^2$$

따라서 늘어난 후의 넓이는

$$\text{선미} = 0.030827464397 + 0.002231852626 = 0.033059317023 \text{ m}^2$$

$$\text{중앙 평행부} = 0.138125 + 0.01 = 0.148125 \text{ m}^2$$

$$\text{선수} = 0.028738217103 + 0.003669204865 = 0.032407421968 \text{ m}^2$$

$$\text{격벽} = 0.084513593955 + 0.007801254827 = 0.092314848782 \text{ m}^2$$

1.7.2.2. 선미, 선수, 중앙 평행부 부분의 Deck의 넓이

선미, 선수, 중앙 평행부 부분의 Deck는 모두 벽쪽의 Deck 넓이만 위로 직사각형 형태로 넓이가 늘어났다고 생각 할 수 있다. 따라서 그 늘어난 넓이를 직사각형 넓이 구하는 공식을 이용하여 계산해보면

선미의 늘어난 넓이 = $0.005249535312 \text{ m}^2$

중앙 평행부의 늘어난 넓이 = 0.0106 m^2

선수의 늘어난 넓이 = $0.003747684941 \text{ m}^2$

따라서 늘어난 후의 넓이는

선미 = $0.07294268084 + 0.005249535312 = 0.078192216152 \text{ m}^2$

중앙 평행부 = $0.173404036836 + 0.0106 = 0.184004036836 \text{ m}^2$

선수 = $0.047497873482 + 0.003747684941 = 0.051245558423 \text{ m}^2$

이다.

1.7.2.3. Z-축 프레임의 넓이

Z-축 프레임의 넓이는 선형이 바뀌지 않았으므로 전 프레임의 넓이와 같다고 볼 수 있다.

따라서 Z-축 프레임의 넓이는 $0.11278315899 \text{ m}^2$ 이다.

1.7.2.4. 용골의 넓이

용골은 단지 처음 모양의 용골이 3개로 늘어난 것 뿐이다.(물론 완전히 똑 같은 모양은 아니지만 근사시키면 같다고 볼 수 있다.)

따라서 원래의 용골의 넓이가 0.00972037 m^2 이므로

용골의 넓이 = $0.00972037 + 0.01944074 = 0.02916111 \text{ m}^2$

1.7.3. 경하중량의 재계산

앞서 구한 것을 살펴보면(단위 : cm^2)

격벽의 넓이 = 923.148

용골의 넓이 = 291.611

프레임의 넓이 = 3263.749

Deck의 넓이 = 3134.418

선형 설계 후 C_p 커브²에서

² 깊이가 늘었음에도 표면의 넓이가 같은 값을 갖는 이유는 배수량은 늘었으나 선체 외판의

L O A =	1.00 M	L/B =	5.4645	LCB =	0.64 % LPP
L B P =	1.00 M	B/T =	1.5250	VCB =	0.06 M
BEAM =	0.18 M	CB =	0.8289	KMT =	0.0879 M
DEPTH =	0.12 M	CBF =	0.8440	LE =	0.200 M(20.00 %)
DRAFT =	<u>0.12 M(TF)</u>	CBA =	0.8138	LX =	0.530 M(53.00 %)
	<u>= 0.12 M(TA)</u>	CM =	0.9861	LR =	0.270 M(27.00 %)
SPEED =	0.00 KTS	CW =	0.9281	AFP =	0.000 M^2(0.00 % AM)
FN =	0.0000	CWF =	0.8796	BL =	0.000 M(0.00 % LPP)
SCH =	0.000 M	CWA =	0.9767	WSA =	<u>0.4 M^2</u>
DIA =	0.000 M	CB/(L/B)=	0.1517	VOL =	0.0 M^3

(흘수를 배의 높이와 같게 맞추어 표면적을 계산하였다. 빨간 밑줄 참고)³

선체의 표면의 넓이 = 4000

수업자료로부터 모은 정보에서

선체의 두께 = 0.2cm	퍼티의 두께 = 0.05cm
에폭시의 두께 = 0.05cm	기관부의 무게 = 1 kg
우드락 밀도 = 0.7 g/cm^3	퍼티의 밀도 = 1.35 g/cm^3
에폭시의 밀도 = 1.5 g/cm^3	

프레임과 Deck의 두께는 0.2cm, 선체와 격벽과 용골의 두께는 0.3cm⁴으로 하여 계산을 해 본 결과

$$4000 \times 0.3 \times 0.7 + 4000 \times 0.05 \times 1.35 + 4000 \times 0.05 \times 1.5 + 923.148 \times 0.3 \times 0.7 + 291.611 \times 0.3 \times 0.7 + 3263.749 \times 0.2 \times 0.7 + 3134.418 \times 0.2 \times 0.7 + 1000 = 3560.84277 \text{ g}$$

앞에서 구한 배수량은 9858.576g 이다. 재화중량을 계산해보면

$$9858.576 - 3560.84277 = 6297.73323 \text{ g}$$

이번에 우리가 건조해야 할 배의 조건인 재화중량 6kg을 상회하는 것을 알 수 있다.

무게는 늘지 않았음을 의미하므로 필요한 재화중량에 미치지 못한 점을 보완해 준다. .

⁴ 배수량을 많이 확보했기 때문에 튼튼하게 할 목적으로 저번보다 두껍게 잡게 되었다.

2. 제어부

2.1. 기본 소자 조사

2.1.1. 마이크로프로세서(MPU)와 마이크로컨트롤러(MCU)

2.1.1.1. 마이크로프로세서란? (Microprocessor)



그림 2.1. 마이크로프로세서

마이크로프로세서(microprocessor, 간단히 " μ P"라고 표기함)는 컴퓨터의 중앙처리장치(Central Processing Unit ; CPU)를 단일 IC(Integrated Circuit) 칩에 집적시켜 만든 반도체 소자로서, 1971년에 미국의 Intel사에 의하여 세계 최초로 만들어졌으며 오늘날은 이를 흔히 MPU(MicroProcessor Unit)라고 부르기도 한다.

즉, 마이크로프로세서는 CPU의 여러 형태중에서 1개의 소자로 만들어진 종류이므로, 모든 마이크로프로세서는 CPU이지만 모든 CPU가 곧 마이크로프로세서인 것은 아니다. 그러나, 이를 MPU라고 부를 때는 이것이 곧 마이크로프로세서를 의미한다. 이것은 일반 컴퓨터의 중앙처리장치에서 주기억장치를 제외한 연산장치, 제어장치 및 각종 레지스터들을 단지 1개의 IC 소자에 집적시킨 것이므로 연산, 제어, 판단, 기억, 계시, 프로그램에 의한 처리 등의 기본적인 처리기능은 일반 중 대형 컴퓨터의 CPU에 비하여 별로 다를 바가 없다.

한편, 이와 같은 마이크로프로세서를 CPU로 사용하여 만든 소형 컴퓨터를 마이크로컴퓨터(microcomputer)라고 한다. 이에 비하여 미니컴퓨터 이상의 중대형 컴퓨터에서는 CPU를 많은 소자를 사용하여 전용으로 설계하게 된다. 마이크로컴퓨터에서처럼 마이크로프로세서를 사용하여 컴퓨터를 설계하면 CPU가 하나의 IC 소자로 간단히 구성되므로 전체 시스템이 소형경량화되고, 소비전력이 적어지며, 가격이 낮아질 뿐만 아니라 부품수가 적어서 시스템의 신뢰성이 높아지는 등의 장점을 갖는다.

이와 같이 마이크로컴퓨터가 대형이나 미니컴퓨터와 구별되는 가장 큰 구조상의 특징은 CPU로서 마이크로프로세서를 사용한다는 것인데, 최근에는 미니컴퓨터급에 마이크로프로세서를 사용하는 경우가 점점 많아지고 있어서 이들의 분류기준이 모호해지고 있다. 또한, 이

처럼 마이크로컴퓨터의 성능이 크게 향상되어 슈퍼마이크로화 함에 따라 미니컴퓨터의 입지는 점점 좁아지고 있다.

2.1.1.2. 마이크로프로세서의 구성

컴퓨터의 연산장치와 제어장치를 1개의 작은 실리콘 칩에 집적시킨 처리장치.

연산을 미리 확립된 순서에 의해 체계적으로 실행할 뿐만 아니라 컴퓨터의 각 장치에 제어 신호를 제공하는 제어장치를 1개의 작은 실리콘 칩에 집적시킨 초대규모 집적회로로 이루어진 처리장치이다.

내부는 산술논리연산기, 레지스터, 프로그램 카운터, 명령 디코더, 제어회로 등으로 구성되어 있다.

마이크로프로세서는 주기억장치에 저장되어 있는 명령어를 인출하여 해독하고, 해독된 명령어를 실행하며 실행 결과를 다시 주기억장치에 저장할 수 있는 기능 등을 자동으로 수행함과 동시에 입출력 장치들과도 데이터 교환을 수행한다.

2.1.1.3. 마이크로프로세서의 발전과 종류

반도체 기술은 지난 20여 년 동안 실로 혁명적이라 말할 수 있을 정도로 발전하여 현재는 대규모 집적회로(LSI)와 초대규모 집적회로(VLSI)의 집적도는 1개의 작은 실리콘 칩에 500만 개 이상의 트랜지스터를 집적할 수 있다. 반도체 집적 기술의 이와 같은 발달로 컴퓨터 시스템의 크기가 작아졌을 뿐만 아니라 시스템의 신뢰도와 동작 속도가 크게 개선되었고 전력 소비도 많이 줄었으며 가격 또한 매우 저렴해졌다.

현재 마이크로프로세서는 마이크로컴퓨터에는 물론 군사용 무기와 일반 가정에서 사용되는 가전제품에 이르기까지 다양하게 이용되고 있으며, 또한 공정 제어 분야와 사무자동화기기 등에도 광범위하게 이용되고 있다. 또한 여러 반도체 회사에서 앞을 다투어 개발하고 있어 성능이 우수하면서 가격이 매우 저렴한 마이크로프로세서가 많이 나와 있음.

마이크로 프로세서는 일반적으로 MPU로 불리며 한개의 LSI로 집적돼 있지만 몇개의 LSI에 분할되는 경우도 있으며 8비트MPU 또는 32비트MPU하는 식으로 비트수와 함께 부르는 것이 보통이다.

MPU에는 종래 소프트웨어로 실행하고 있던 기능과 명령을 하드웨어로 실행하기 위한 복잡한 명령세트를 갖는 CISC(Complex Instruction Set Computer), 가급적 간단한 명령세트만을 사용해 하드웨어를 단순화한 RISC(Reduced Instruction Set Computer), 4비트 또는 2비트의 MPU를 몇개씩 접속해 비트를 사용자의 희망에 맞춘 비트스라이스MPU 등이 있다. 마이크로프로세서를 처음 개발한 것은 미국의 인텔사로 4비트의 MPU인 4004를 1971년에 개발했다.

2.1.1.4. 마이크로프로세서의 역사

현재 널리 사용되고 있는 마이크로프로세서는 인텔사의 80계열과 모토롤라사의 68계열이

있다. 1969년 최초로 발표된 4004는 4비트를 처리할 수 있는 마이크로프로세서였다. 그러나 마이크로프로세서가 오늘날처럼 대중적으로 사용되기 시작한 것은 인텔사의 8비트 마이크로프로세서인 8080이 나온 후부터이다. 8080은 특히 CP/M 운영체제의 등장과 함께 개인용 컴퓨터의 CPU로 매우 널리 사용되었으며, 자일로그사에서 8080을 개량하여 만든 Z80은 오늘날까지도 많이 사용되는 8비트 마이크로프로세서가 되었다. 한편 모토롤라사의 6800과 6809, 모스텍사의 6502 등도 널리 사용된 8비트 마이크로프로세서였다.

1970년대 후반부터 16비트 마이크로프로세서가 등장하였다. 인텔사의 8086과 모토롤라사의 68000은 초창기의 16비트 마이크로프로세서인데, 이들은 각각 IBM PC와 매킨토시 마이크로컴퓨터에 채용되어 널리 알려졌다. 인텔사의 80386과 80486, 그리고 모토롤라사의 68020과 68030, 68040은 모두 근래에 나온 32비트 마이크로프로세서이며, 이들의 성능은 80년대 초의 미니 또는 그 이상의 컴퓨터를 능가한다. 특히 80486과 68040 이상의 마이크로프로세서들은 하나의 실리콘 칩에 100만개 이상의 트랜지스터들을 직접한 고성능의 제품이다.

응용 면에서 보면 8비트 마이크로프로세서는 주로 가정용의 PC나 게임기, 가전제품이나 기계 제어 등에 사용되고 있고, 16비트 마이크로프로세서는 일반적인 개인용 컴퓨터나 산업용의 제어기기에, 32비트 마이크로프로세서는 고성능의 개인용 컴퓨터나 워크스테이션, 미니급 컴퓨터에 주로 사용되고 있다. 특히 근래는 미니 컴퓨터나 대형 컴퓨터를 사용하기보다는 값싼 마이크로프로세서들을 여러개 연결하여 빠른 처리 속도를 내게 하는 병렬 처리(Parallel processing)에 대한 연구가 많이 이루어져 있다.



그림 2.2. AVR

2.1.1.5. 마이크로컨트롤러란?

마이크로프로세서 중에 1개의 칩내에 CPU 기능은 물론이고 일정한 용량의 메모리(ROM, RAM 등)와 입출력 제어 인터페이스 회로까지를 내장한 것을 특별히 마이크로컨트롤러라고 부른다. 이것은 일반 마이크로프로세서가 널리 사용되기 시작하면서 1970년대 중반부터 나타난 형태로서 범용의 목적보다는 기기 제어용에 주로 사용되므로 붙여진 이름이며, 이는 1개의 소자만으로 완전한 하나의 컴퓨터 기능을 갖추고 있으므로 "단일 칩 마이크로컴퓨터(one-chip 또는 single-chip microcomputer)"라고도 부른다.

2.1.1.6. 마이크로컨트롤러의 특징

제어 응용 시스템에 마이크로컨트롤러를 사용하게 되면 범용 마이크로프로세서나 하드웨어 논리를 사용하는데 비하여 다음과 같은 장점을 얻을 수 있다.

① 제품이 소형경량화 된다. 시스템의 컨트롤러 부분이 마이크로컨트롤러 LSI 1개와 극히 소수의 외부 소자들도 간단히 구성되므로 크기와 무게가 현저히 줄어들고 소비전력도 적어진다. 이에 따라 부수적으로 전원장치까지도 소형 경량화된다.

② 제품의 가격이 싸진다. 컨트롤러 부분이 단순화됨에 따라 부품비, 제작비, 개발비가 감소되고, 개발기간도 단축된다.

③ 시스템의 신뢰성이 향상된다. 컨트롤러가 단순화되어 부품수가 적어지고 신뢰도가 높은 소자를 사용하므로 고장률이 감소하며 유지보수가 용이해진다.

④ 시스템의 융통성이 커진다. 하드웨어에 의존하는 부분을 소프트웨어로 처리할 수 있게 되므로 기능의 변경이나 확장에 보다 유연하게 대응할 수 있다.

2.1.1.7. 마이크로컨트롤러의 발전

종래의 마이크로컨트롤러는 CPU 기능에다 한정된 용량의 메모리(기본적으로 SRAM을 가지고 있으며 EPROM, OTP ROM, 또는 플래시 메모리를 함께 내장하는 모델도 있음)와 인터럽트 제어기, 직렬 및 병렬 인터페이스, 카운터/타이머 등을 내장하는 정도에 불과하였으나, 최근에는 DMA 제어기와 같은 고성능의 I/O 기능, 고속의 직렬통신 기능, A/D 컨버터, PWM 출력, PLL 회로 등을 내장하여 각종 제어 시스템에 사용하기에 보다 편리해지고 있다.

이와 같이 오늘날의 마이크로컨트롤러는 각 응용분야에 적합하도록 매우 많은 종류와 모델이 개발 사용되고 있으며, 범용 마이크로프로세서를 지칭하는 MPU와 구별하여 MCU(MicroController Unit 또는 MicroComputer Unit)라고 부르기도 한다.

2.1.1.8. 마이크로컨트롤러의 종류

인텔(Intel)에서는 80486, Pentium 등 일반 PC를 위한 CPU 이외에 산업용으로 MCS(Micro Control System) 계열의 CPU를 생산해 오고 있다.

MCS-48, MCS-51, MCS-251, MCS-96이 그것이다. 그 중에 8051을 코어(core)로 하는 마이크로 컨트롤러들을 통칭해서 MCS-51 패밀리(family)라고 부른다.

마이크로프로세서(Microprocessor)라 불리는 CPU는 보조해주는 주변 LSI 칩들이 필요하다. 예를 들어, Z80을 컨트롤러로 사용하기 위해서는 CTC(타이머), PIO(병렬 입출력), SIO(직렬 입출력), ROM(프로그램 메모리), RAM(데이터 메모리) 등이 기본적으로 필요하다. 그러나, 마이크로 컨트롤러인 8051의 경우 이들 CPU를 보조하는 LSI들을 칩에 내장하고 있기 때문에, 사용자가 이를 구성할 필요가 없어 보다 편리하고 간단하게 컨트롤러를 구성할 수 있다.

MCS-51은, 기본 구조는 같으면서 여러 가지로 기능을 다양화하여 IC 화해 놓아서, 사용자가 필요한 것을 고를 수 있는 폭이 넓다. 기본적인 8051 이외에 8051에는 들어 있는 내부 롬(ROM)이 없는 8031, 몇 가지 기능이 확장된 8052, 내부 롬이 EPROM으로 되어 있는 8751 등이 나와 있다.

2.1.1.9. 마이크로컨트롤러의 응용(활용)사례

핸드폰, MP3플레이어, 디지털카메라, DVD플레이어, 셋톱박스, 전화기, 냉장고, 전자레인지, 세탁기, 인공위성, 자동차, 선박, 항공기, GPS, 의료기기, 계측장비 등 가전-의료-산업분야에 두루 사용된다. 뿐만 아니라 최근에는 중공업 산업로봇, 반도체 검사용 비전로봇, 원자로 상태 검사로봇 등으로 상용화된 로봇에도 이용되었다.

2.1.1.10. 마이크로프로세서와 마이크로컨트롤러의 차이점.

MPU : Micro processor unit

└ 두가지 모두 CPU, 처리하는 과정은 같지만 개념적으로 어

MCU : Micro controller unit ┐ 떤 부분에 비중을 두는가에 따라 구분됨.

CPU는 ALU에서의 연산과정을 통해 명령을 수행, 결과값을 주기억장치(RAM)에 저장하게 된다. 이에 따라 CPU칩의 구조방식에 차이가 있는데, Off chip과 On chip개념으로 나뉘게 된다.

┐ Off chip : CPU와 주기억장치, I/O Port가 외부에 존재하게 됩니다. MPU

└ On chip : 내부에 모두가(CPU,주기억장치,I/O Port) 집적되어있다. MCU

MPU는 오프칩 형태로 이뤄져 있는데, Processor말 그대로 연산을 하는데, 무엇보다도 큰 비중을 두게 된다. 사칙연산, 논리연산, 보수 등을 다루게 되는 CPU로써 더욱더 큰 메모리를 요구하게 되고, 연산처리에 하는 레지스터로써 결과값을 저장하는 어큐뮬레이터(Accumulator)의 비중이 커지게 된다. 한마디로 MPU는 연산이 주목적인 CPU다. 이러한 연산중심의 CPU는 오프칩 형태로 연산처리에 주변장치가 특히 주기억장치의 비중이 크기 때문에 오프칩 형태를 쓴다. 그래서 컴퓨터의 램이 외부에 장착되어 있다.

이 CPU는 보통 컴퓨터에서 쓰이는 중앙처리장치를 말하고, 연산처리에 필요한 주기억장치가 크다는 것과 CPU의 메모리맵이 isolate방식으로 외부에 포트나 부가적 기능이 따로 분리되어 있는 독립된 형태가 주를 이루고 있습니다.

이에 반하여 MCU는 연산이 목적이 아닌 말 그대로 컨트롤 목적이므로 Off chip형태가 아닌 통합된 칩인 On chip형태를 이루고 있다.

컨트롤러는 사용자가 직접컨트롤 하여, 메모리맵 구성하며 User의 필요한 기능만을 부가적으로 셋팅하여 쓸 수 있는 형태로 되어있다. 온칩 형태이기 때문에 오프칩의 메모리맵과

는 다른 형태로 메모리영역이 줄어들게 되어, 데이터 처리에 있어 제한된 기능만을 수행하는 형태가 될 수밖에 없다.

앞서 말한바와 같이 MCU는 필요한 기능을 컨트롤 한다는 목적으로 사용되고, 이러한 칩에는 고유한 기능을 가지고 있으며, 사용자의 필요에 따라 각각 다른 수십 가지의 칩 기능을 가지고 있다. 이러한 ON chip형태는 집적화로 인해 좁은 공간의 회로에서도 많은 기능을 수행할 수 있고, CPU내부에 메모리가 있어 명령에 필요한 프로그램도 집어넣을 수 있다. 이러한 칩은 4bit, 8bit, 16bit, 32bit형태로써 데이터를 처리하는 속도를 가지고 있으며 수많은 가전제품의 자동제어장치 시스템이나, 도난 방지시스템, 자동차ABS시스템, 항온 시스템 등 등 다른 분야에서도 널리 쓰이고 있는 CPU라 할 수 있다.

마이크로컨트롤러는 마이크로프로세서에다가 메모리와 입출력포트,타이머,카운터,인터럽터,시리얼포트 등을 단일(하나로)칩에 구현한 마이크로컴퓨터(=마이컴)라 부르며 마이크로컨트롤러라고도 한다. 요약하자면 보조칩이 있는가 없는가의 차이이다.

간략히 말하면, MPU는 컴퓨터용CPU로 연산목적(Pin-out으로 CPU와 메모리,I/O가 분리된 형태)이고, MCU는 가전제품이나 특히 널리 잘 알려진 로봇(마이크로마우스, 2족 로봇)에 많이 쓰이는 User controller CPU(chip에 메모리와 I/O가 내장)라 할 수 있다.

2.2. AVR 조사

2.2.1. AVR 이란 무엇인가?

AVR

=> Alf(Bogen) Vergard(Wollan) Risc

ATMEL사에서 제작된 8bit CMOS 구조의 RISC Micro Controller

<용어정리>

1.RISC(reduced instruction set computer): 범용 마이크로프로세서의 명령세트를 축소하여 설계한 컴퓨터

-복잡한 처리는 소프트웨어에게 맡기는 방법을 택한 컴퓨터

2.CISC(complex instruction set computer)

소프트웨어 특히 컴파일러 작성을 쉽게 하기 위해 하드웨어화 할 수 있는 것은 모두 하드웨어에 맡기는 방법을 택한 컴퓨터

♣RISC의 특징을 CISC와 비교

-명령의 대부분은 1머신 사이클에 실행되고, 단순하다.

-레지스터 수가 많으며 마이크로프로그램을 저장하는 칩의 공간에 레지스터 배치

-컴파일러의 최적화 필요함. (파이프라인의 효과적 사용)

3. MISP(million instructions per second)

1초당 100만 개 단위의 명령어 연산이란 뜻으로 컴퓨터의 연산속도를 나타내는 단위

-16MISP 라는 것은 1초에 1600만개의 명령어를 처리한다는 것임.

2.2.1. AVR의 종류

1) ATtiny 패밀리

꼭 필요한 최소한의 기능만들을 뽑아서 아주 작게 만든 칩

-소형시스템에 많이 쓰이기 때문에 1KB~2KB의 플래시 메모리

-PIN 수: 8핀, 20핀, 32핀

-외부 메모리 확장 불가능

-가격이 낮고 기능이나 성능도 낮다.

2) AT90 패밀리

ATtiny 패밀리보다 약간 더 복잡한 사양으로 만들어진 칩

-1KB~8KB의 플래시 메모리

-PIN 수: 8핀, 20핀, 28핀, 32핀, 40핀, 44핀

-외부 메모리 확장 가능

-내부 메모리 사이즈가 크기 때문에 C 언어로도 프로그래밍 가능

3) ATmega 패밀리

가장 규모가 크고 성능이 높은 응용분야에 사용

-8KB~128KB의 플래시 메모리

-PIN 수: 28핀, 32핀, 40핀, 44핀, 64핀

- 내부데이터 메모리:EEPROM(256-4096Byte),SRAM(512-4096Byte)
- 가격도 높고 기능이나 성능도 높다.
- 모델명: ATmega8, ATmega16, ATmega32, ATmega64, ATmega128
(분류기준: 저장하는 플래시 메모리 용량에 따라)

2.2.2. AVR의 특징

PIC보다 처리속도가 빠르고 8051보다 자원이 풍부하다라는 특징이 있어
중소 규모의 산업용 제어기 제작에 적합하다.

1) RISC 구조의 고속 처리가 가능하다.

- 같은 클럭으로 동작시 PIC보다 4배 빠르고, 8051보다 10배 빠르다.
1MHz에 1MIPS의 처리 능력이 있어1MHz로 동작하는 경우 10MIPS의 처리 능력 을 발휘한다.

2) C 언어의 처리 강력지원

- CPU 설계 단계에서 레지스터/메모리/명령어가 C언어에 적합하도록 설계
C 언어를 사용하면 개발기간이 단축 및 유지보수 편리

3) ISP기능을 하는 FLASH와 EEPROM이 있어 AVR을 기판에 부착한 상태에서 여러번 프로그래밍 할 수 있음

4) 다양한 AVR 제품군이 있어 용도에 적합한 소자를 선정하여 사용할 수 있다.

5) 풍부한 내부 자원을 구비하고 있어 외부와 별도로 주변장치를 부착하지 않아도 된다.

AVR은 개발단계부터 C언어 사용을 고려하여 제작되었으므로, 코드생성 면에서 PIC나 8051 보다 유리함.

2.2.3. AVR의 개발환경

AVR은 내부의 플래쉬 메모리와 EEPROM에 데이터를 써넣을 수 있도록 SPI 인터페이스를 제공한다. SPI는 오직 3라인을 이용한 통신 방법으로 MOSI(Master Out Slave In), MISO(Master In Slave Out), SCLK(SPI CLOCK) 시그널을 이용한다. Motorola에서 개발되었으며 Master와 Slave가 SCLK에 동기하여 데이터를 교환하는 방식이다. 플래쉬 메모리를

엑세스 하기 위해서는 AVR의 Reset 핀을 low로 한 상태에서 앞의 세 시그널을 이용하여 데이터를 읽고 쓰기가 가능하다. 즉, AVR과의 SPI 인터페이스를 맞추어주기만 하면 내부 플래쉬 메모리의 엑세스가 가능하다는 것이다. 이는 AVR의 프로그래머를 쓴 값으로 구현하게 하는 동기가 된다.

2.3. ATMEGA 8535 조사

2.3.1. ATMEGA 8535의 특징

- ◆ AVR RISC(Reduced Instruction Set Computer) 구조를 사용
- ◆ 고속 수행, 저전력 소모용 RISC 구조 설계
 - 118개의 강력한 명령구조
 - 대부분 단일 클록 사이클에 의한 실행가능
 - $32 * 8$ 범용 워킹 레지스터(Working Registers)
 - 10MHz에서 최대 10MIPS의 처리 속도를 갖는다.
- ◆ 데이터와 비휘발성 프로그램 메모리 구조
 - 프로그램 가능한 8KByte의 플래시 메모리 내장(8KBytes of In-System Programmable Flash)
 - 최대 프로그램 가능 횟수 : 약 10000번 write/erase 가능
 - 512 Byte 의 SRAM
 - 512 Byte 의 프로그램 가능한 EEPROM 내장 (최대100000번write/erase 가능)
 - 플래시 프로그램과 EEPROM 데이터 보호용 Lock 기능
- ◆ 주변 장치의 특성(Peripheral Features)
 - 별도의 프리스케일러가 있는 1개의 8bit 타이머/카운터 내장/16bit 타이머/카운터 내장
 - 비교,캡처모드와 8-,9-,10-비트 PWM 기능 내장
 - 아날로그 비교기 내장 (On-chip Analog Computer)
 - 오실레이터가 내장되고 프로그램도 가능한 워치도그 타이머
 - 시스템 내부 프로그래밍용 SPI 직렬 인터페이스 방식
 - 전이중 방식의 UART(시리얼통신포트)
- ◆ 전용 마이크로컨트롤러 특성
 - 저전력 아이들(Idle : 휴식) 모드와 전력 절약모드
 - 외부와 내부 인터럽트 소스

2.3.2. ATMEGA 8535의 구조

ATmega8535는 AVR RISC 구조를 기본으로 하는 전전력 CMOS 8비트 마이크로컨트롤러이다

단일 클럭 사이클 내에 강력한 명령이 수행되므로 ATmega8535는 8MHz에서 8MIPS의 속도를 낼 수 있다.

AVR은 32개의 범용 작업 레지스터를 이용하여 명령을 처리한다.

모든 32개의 레지스터가 ALU와 직접적으로 연결되어 있고 두 개의 레지스터 연산을 하나의 명령으로 처리할 수 있다.

결론적으로, ATmega8535는 CISC(Complex Instruntion Set Computer) 마이크로컨트롤러보다 10배 가량 빠른 효과적인 코드를 제공한다.

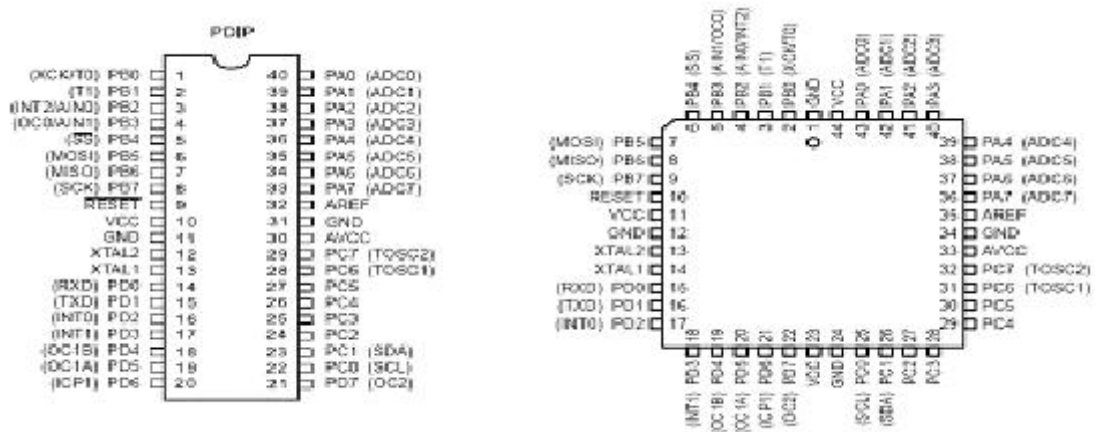


그림 2.3. Atmega8535

Port A(PA7..PA0) ADC로 입력되는 아날로그 입력 핀 (ADC로 사용되지 않을 때에는 8개의 양방향 I/O Port)

Port B(PB7..PB0) 8개의 양방향 I/O Port이다. 내부적으로 풀업 저항 제공. PB2, PB3은 아날로그 컴퍼레이터로 쓸 수 있다./ PB0, PB1은 Timer/Counter 1, 2 의 Source.PB4, PB5, PB6, PB7은 Flash ROM과 EEPROM에 프로그래밍과 직렬 다운로드 포트 이용

Port C(PC7..PC0) 8개의 양방향 I/O Port이다. 내부적으로 풀업저항 제공. PC0는 Two-wire Serial Bus Clock Line이고 PC1은 Two-wire Serial Bus Data I/O line이다. PC6/PC7은 Timer Oscillator 핀이다.

Port D(PD7..PD0) 8개의 양방향 I/O Port이다. 내부적으로 풀업 저항이 제공된다. PD0는 USART의 수신 신호이고, PD1은 USART의 전송신호이다. PD2/PD3은 외부 인터럽트 0/1의 입력 핀이다.

AVR의 Port는 출력이 “0”일 때 포트 핀과 연결된 부하로부터 전류를 20mA 흡수할 수 있으며 출력이 “1”일 때 포트 핀과 연결된 부하로 전류를 4mA 공급할 수 있다. 포트의 출력 용량에 대한 설명은 앞으로 우리가 예제를 통해 하드웨어를 설계하거나 마이컴을 설계할 때

에 굉장히 중요한 부분으로 적용된다.

ATMEGA 8535의 내부 및 메모리 구조(메모리 맵)

프로그램 메모리(Flash ROM), 데이터 메모리(SRAM), Data EEPROM, 작업 레지스터, I/O 레지스터로 구성

프로그램 메모리(Flash ROM) : 전원인가와 동시에 실행되는 프로그램을 저장하는 메모리
데이터 메모리(SRAM), 작업 레지스터, 그리고 I/O 레지스터 : CPU 처리시 사용하는 임시 저장용 RAM

Data EEPROM : 전원을 꺼도 지워지지 않는 데이터를 저장

32개 레지스터 중 6개 : 16-Bit로 세 개의 데이터 영역 어드레싱을 위한 간접 레지스터 포인터로 사용 가능

ALU : 레지스터간 산술 및 논리 연산의 처리나, 상수와 레지스터간의 산술 및 논리 연산의 처리 지원

레지스터 오퍼레이션에 덧붙여 편리한 메모리 어드레싱 모드도 또한 레지스터 파일을 사용 지원

I/O 메모리 영역은 64개의 어드레스를 갖는데, 제어 레지스터로의 CPU 주변 기능, 타이머/카운터, A/D 컨버터, 다른 I/O 기능과 관련된 것이다.

AVR은 Harvard Architecture이다. 프로그램과 데이터의 메모리와 버스가 분리된 구조로 프로그램 메모리는 2 단계의 파이프라인으로 액세스 된다. 한 명령이 수행되는 동안 다음 명령은 프로그램 메모리에서 미리 Fetch된다. 이런 개념이 매 클럭 사이클마다 명령을 수행하는 것이 가능토록 하는 것이다. 프로그램 메모리는 시스템 내부의 프로그램 가능한 Flash 메모리이다.

인터럽트나 서브루틴을 호출하는 동안, 프로그램 카운터(PC)의 리턴 어드레스는 스택에 저장된다. 스택은 일반적인 데이터 SRAM안에 두는 것이 효율적이며 스택의 크기는 전체 SRAM의 크기와 SRAM의 사용에 의해서만 제한된다. 모든 사용자 프로그램은 리셋 루틴 안에서 서브루틴이 호출되거나 인터럽트가 발생하기 전에 스택 포인터를 초기화 해야 한다. 8-Bit Stack Pointer는 I/O 영역에서 읽기/쓰기로 접근이 가능하다.

2.3.3. ATMEGA 8535의 메모리 구조

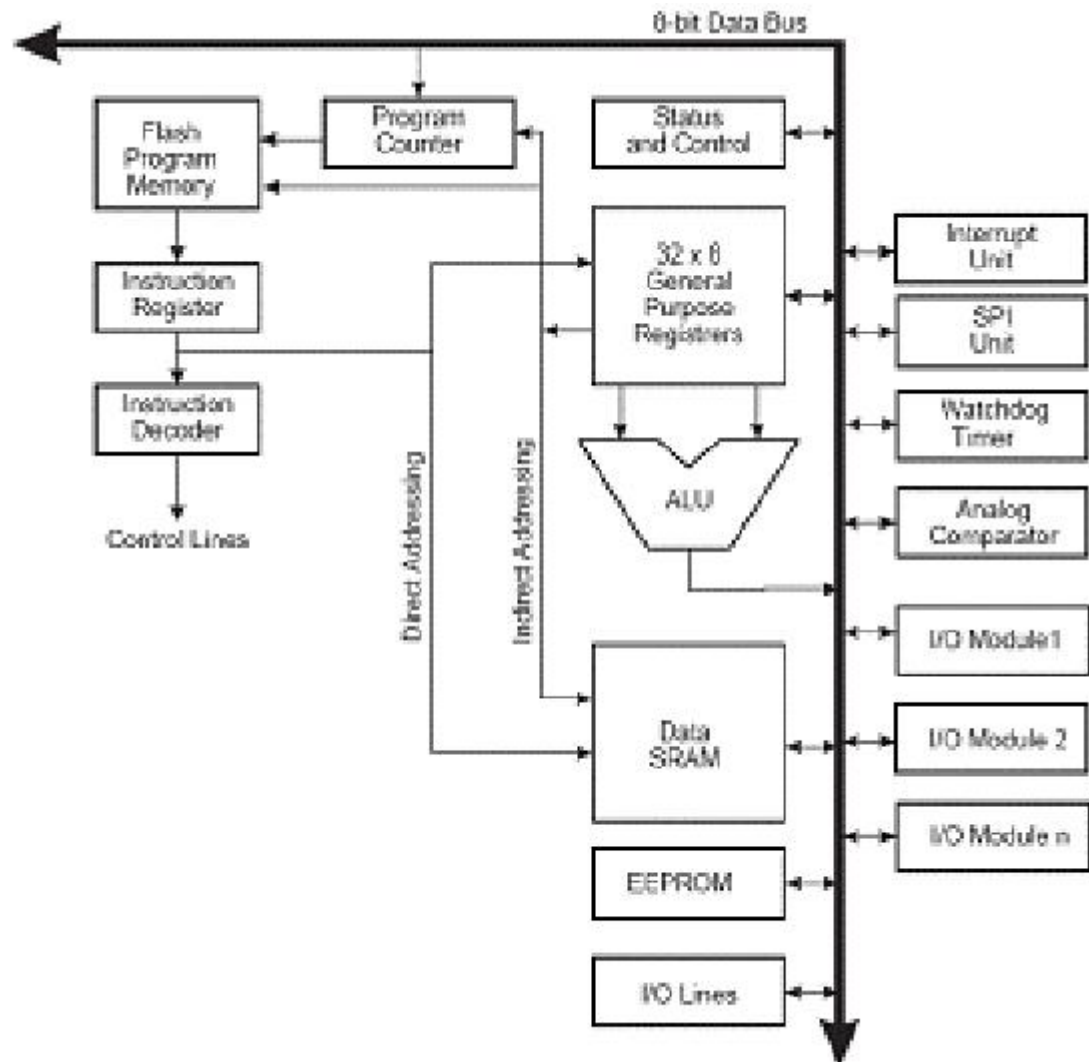


그림 2.4. ATMEGA8535의 메모리 구조

<Controller Diagram>

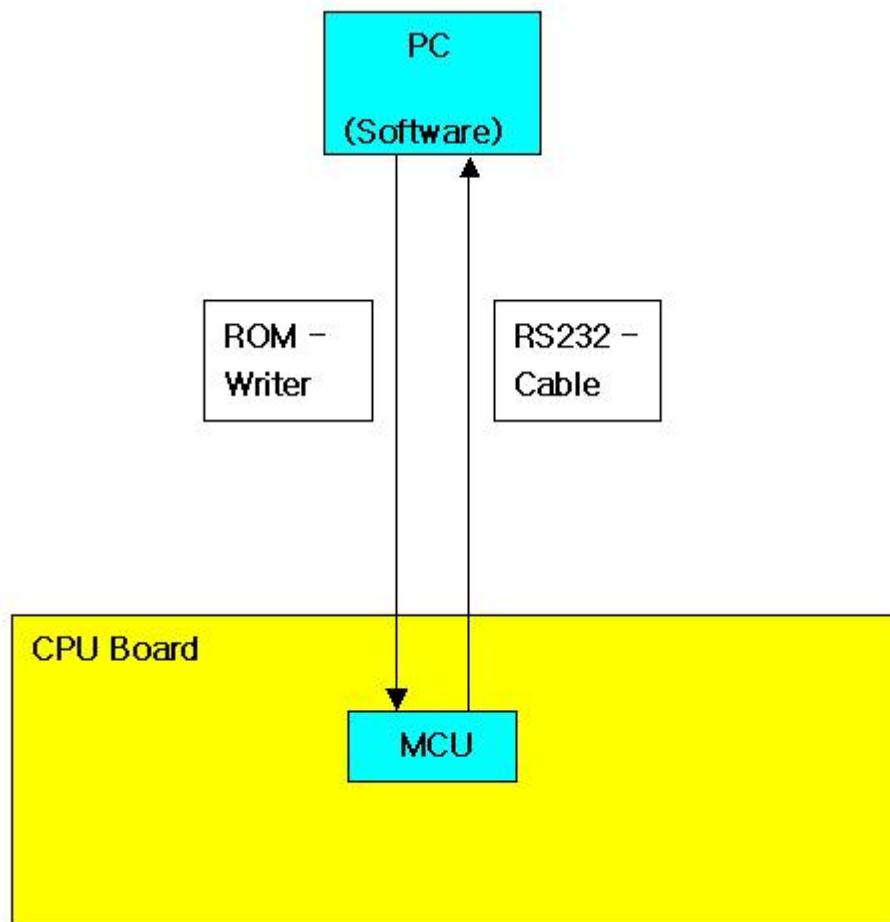


그림 2.5. MCU와 PC간의 정보교환 모식도

MCU(Micro Controll Unit)

CPU-Board : MCU가 동작하도록 꾸며준 회로(각종 전기 소자로 구성)

ROM-Writer : PC에서 작성한 Software를 MCU에 기록하는 장치

RS232-Cable : PC와 Processor간의 데이터 통신을 하는 장치

2.4.2. cpu 보드의 회로도

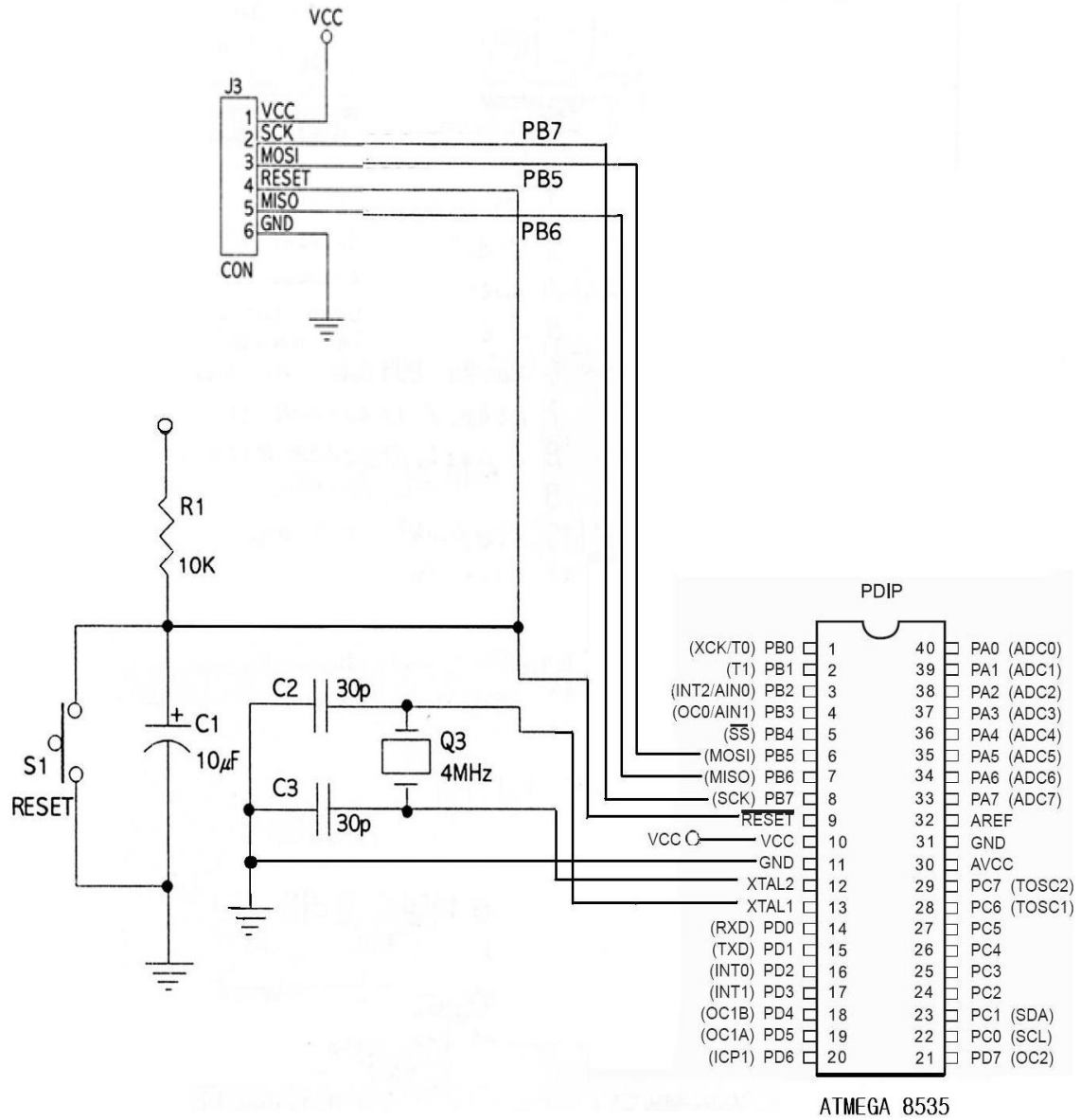
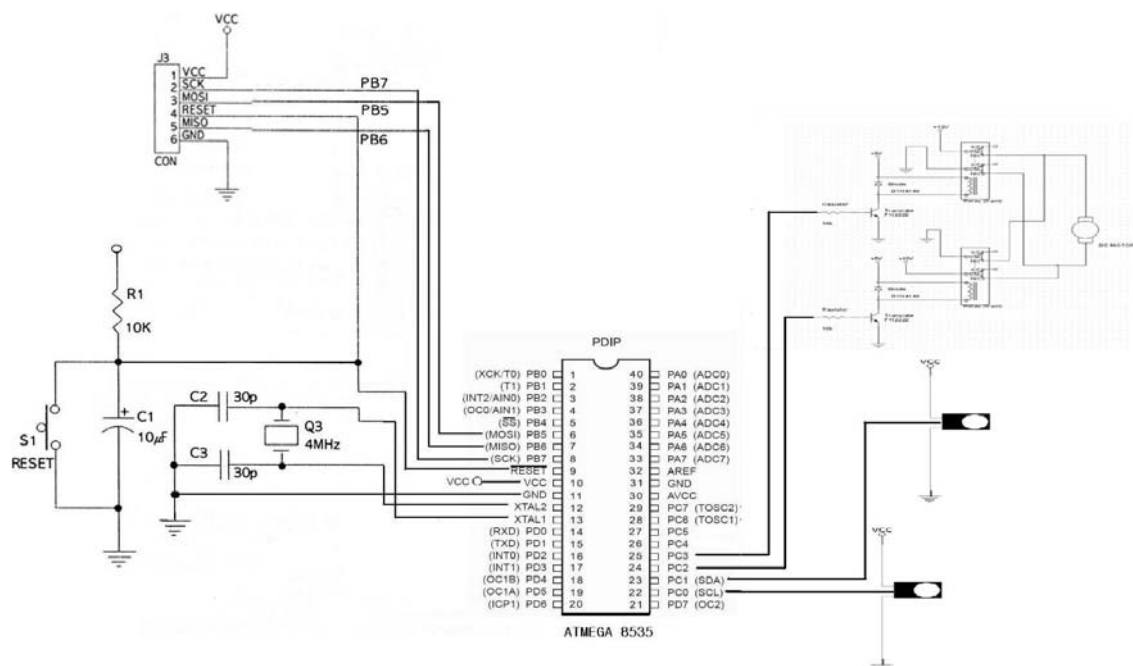


그림 2.7. CPU 보드의 회로도

[illegible]

2.4.4. 서보모터와 DC모터의 회로도



40

<참고 자료> parallel port 핀 내용

Pin No (D-Type 25)	Pin No (Centronics)	SPP Signal	Direction In/out	Register	Hardware Inverted
1	1	nStrobe	In/Out	Control	Yes
2	2	Data 0	Out	Data	
3	3	Data 1	Out	Data	
4	4	Data 2	Out	Data	
5	5	Data 3	Out	Data	
6	6	Data 4	Out	Data	
7	7	Data 5	Out	Data	
8	8	Data 6	Out	Data	
9	9	Data 7	Out	Data	
10	10	nAck	In	Status	
11	11	Busy	In	Status	Yes
12	12	Paper-Out PaperEnd	In	Status	
13	13	Select	In	Status	
14	14	nAuto-Linefeed	In/Out	Control	Yes
15	32	nError / nFault	In	Status	
16	31	nInitialize	In/Out	Control	
17	36	nSelect-Printer nSelect-In	In/Out	Control	Yes
18 - 25	19-30	Ground	Gnd		

그림 2.10. Parallel port 핀

2.4.5. 우리 팀 프로젝트와 관련하여 주어진 과제 (LED 점등 회로도)

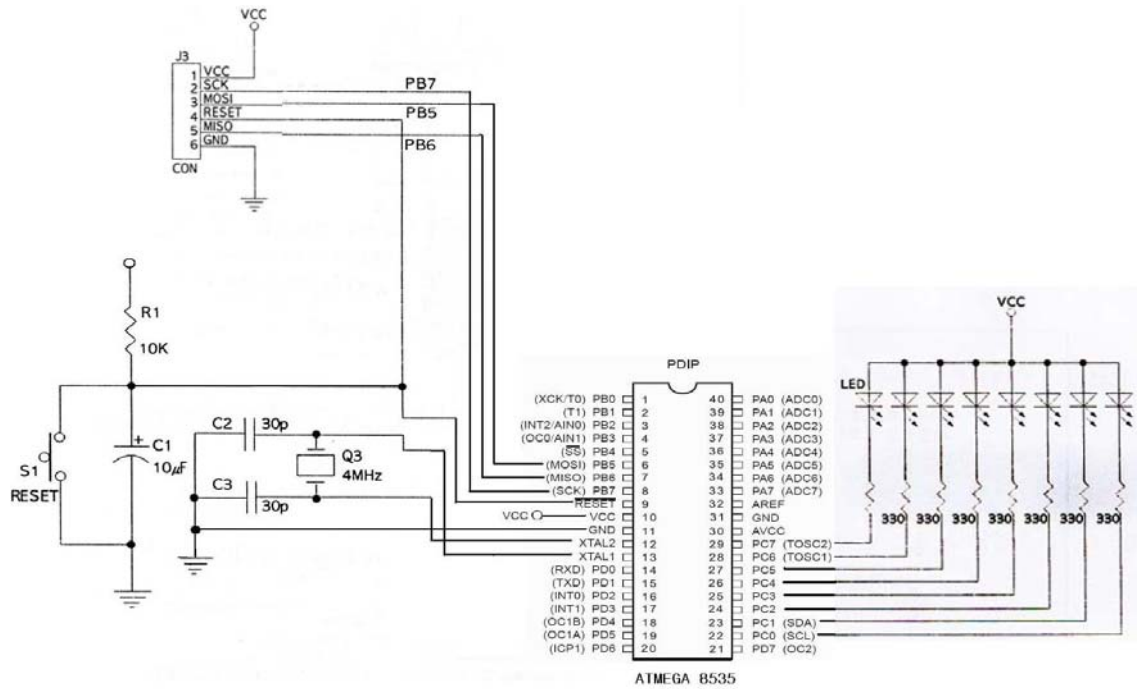


그림 2.11. LED 점등 회로도

2.4.6. 센서연결을 위한 AVR 회로도

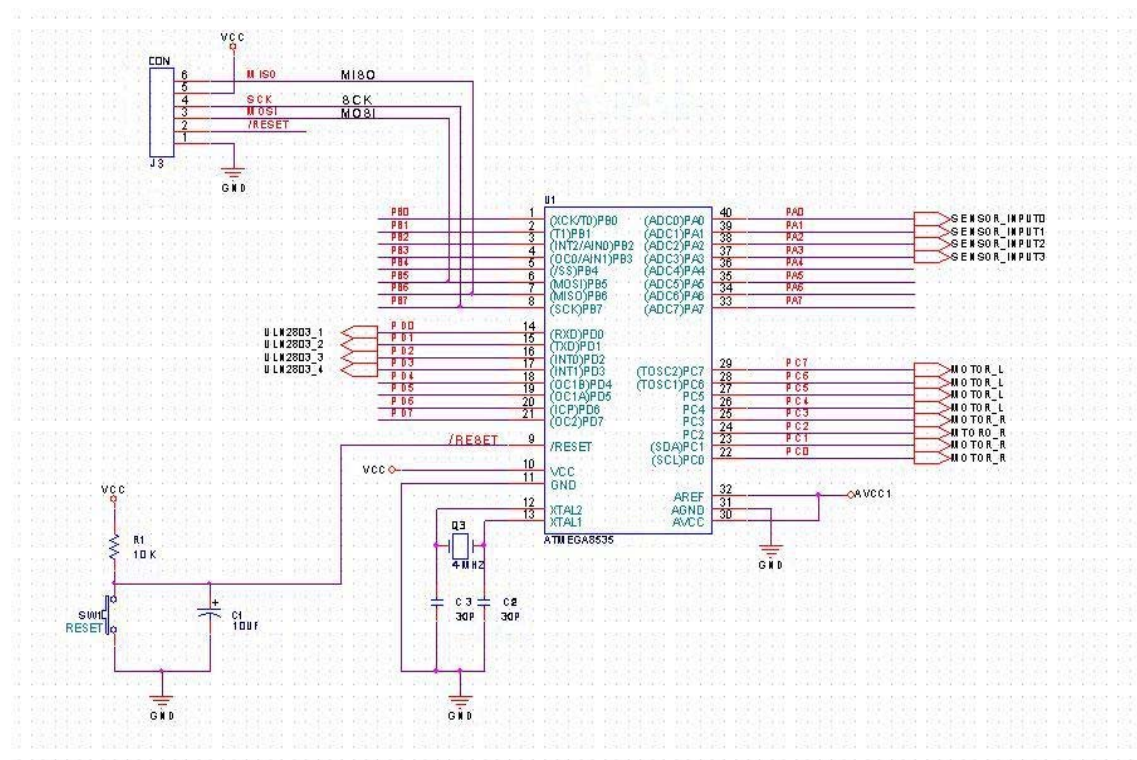


그림 2.12. 센서연결을 위한 AVR 회로도

2.5. 원격 조정부 (수동 조정부)

2.5.1. 기본 개념 조사

1) 진폭변조 (Amplitude Modulation : AM)

변복조 시스템을 논할 때 가장 기본이 되는 진폭변조 (AM)시스템은 구조가 간단하고 구현하기도 쉽기 때문에 여러 분야의 통신에 광범위하게 사용되어 왔다. <그림 2-1-1>에서 보듯이, AM은 신호파형이 변화하는 양을 높은 주파수의 반송파(carrier)의 조밀한 한파장 단위 높낮이에 맞추어 실어보내게 된다.

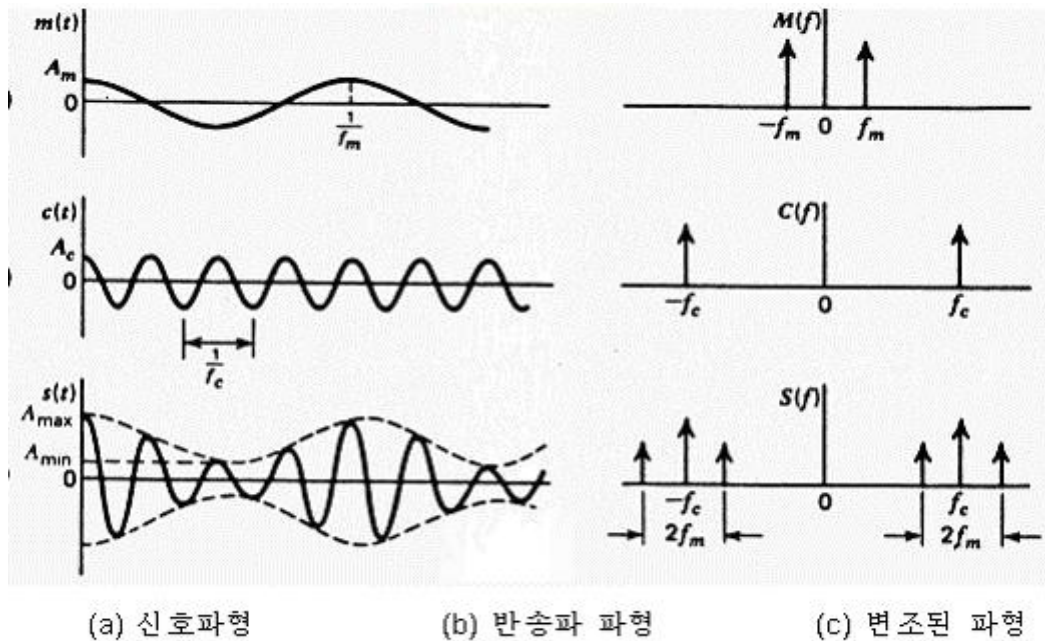


그림 2.13. DSB AM 파의 변조과정

2) DSB (Double Side Band)

일반적인 진폭변조 시스템(Conventional AM)의 경우, <그림 2-1-1>의 주파수 분포도에 서 알 수 있듯이 신호파형이 원점을 중심으로 음과 양의 주파수로 나누어져 있다. 그래서 높은 주파수에 변조시키면 (c)의 주파수 분포도와 같이 반송파 주파수를 기준으로 양쪽에 주파수 성분을 가지게 된다. 이러한 일반적인 AM 변조방식을 DSB(Double Side Band) AM 이라고 한다. 그중에서 변조지수를 높이고 반송파의 전력을 억제하여 효율을 증가시킨 것을 DSB-SC (DSB-Suppressed carrier)라고 한다.

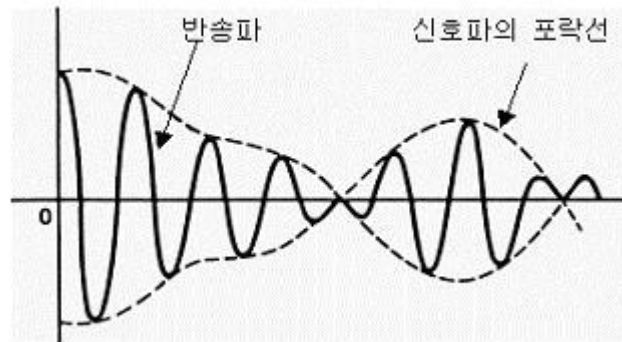


그림 2.14. DSB-SC AM 파의 변조파형

3) 변조지수

변조 시스템에는 변조의 정도나 효율을 나타내는 변조지수 m 이 있다. <그림 2-1-3>에서는 변조지수에 따른 AM 변조파형의 변화를 보여준다. 변조지수는 신호파와 반송파의 전압변화의 관계가 잘 융화되어1에 약간 모자른 것이 효율적이다.

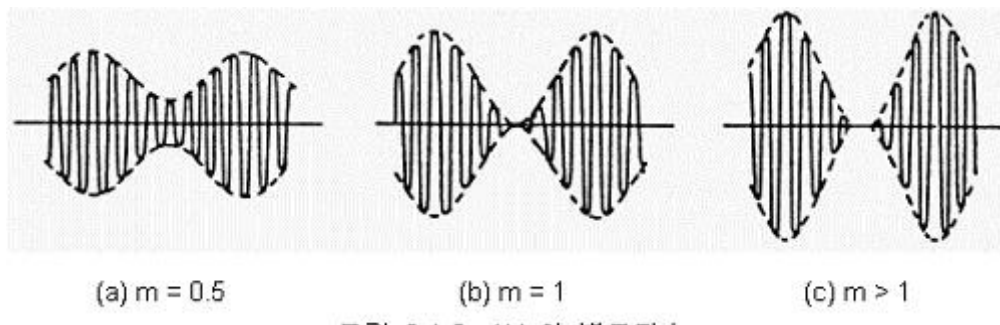


그림 2.15. AM의 변조지수

4) 주파수 변조 (FM ; Frequency Modulation)

AM과 더불어 널리 사용되는 변조방법으로서, 신호파의 크기변화를 반송파의 주파수 변화에 담아서 보내는 방법이다. AM이 주파수는 고정되고 진폭이 변화한 반면, FM은 주파수가 변하는 대신 진폭은 항상 같은 값으로 유지된다. <그림 2-3-1>에서 보여지듯이 신호파형의 전압이 높을수록 주파수가 높아져서 파장이 조밀해지고, 그 반대로 전압이 낮을 때는 주파수가 낮아져서 파장이 넓어지게 된다.

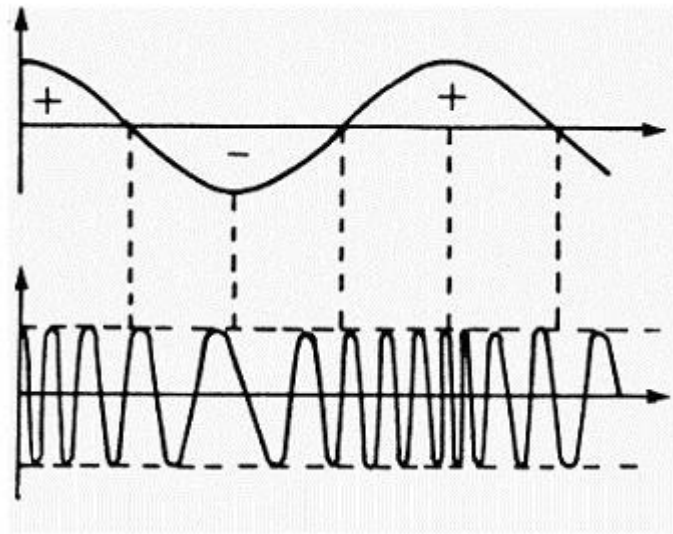


그림 2.16. FM 변조의 원리

5) 변조지수

FM에서의 변조지수 는 다음과 같은 식에 의해 결정된다.

$$\beta = \frac{\text{최대 주파수편이}}{\text{신호의주파수}}$$

이론상 FM통신을 위해서는 무한대의 주파수대역이 필요하지만, 값을 기초로 하여 주요한 주파수 대역만을 선택하여 골라낼 수 있다. 값이 높아질수록 변조율은 높아지겠지만 주파수 대역폭이 점점 더 늘어나게 되며 에 따른 중요 측파대 주파수의 개수는 다음의 표와 같다.

변조지수(β)	0.1	0.3	0.5	1.0	2.0	5.0	10	20	30
중요 측파대 주파수개수	2	4	4	6	8	16	28	50	70

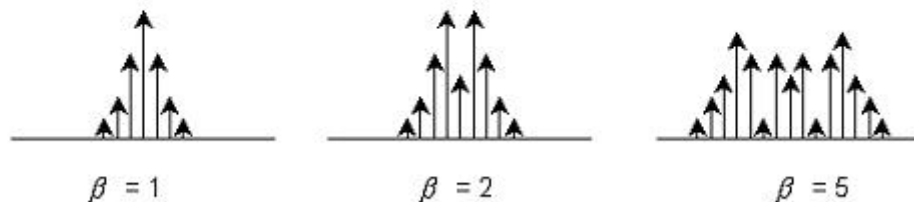


그림 2.17. 변조지수에 따른 FM의 주파수분포

2.6. 크리스탈

2.6.1. 기본개념 조사

1) 수정진동자 [水晶振動子, quartz vibrator]

수정을 그 결정축에 대해 특정한 방향으로 절단하여 만든 얇은 조각의 양면에 도체(導體) 전극을 단 것

여기에 전압을 걸면 전기일그러짐효과에 의해 변형력이 더해져서 진동이 일어난다. 진동이 일어나면 압전효과(壓電效果)에 의해 전극에 전압이 유발된다. 그 진동수는 수정의 역학적 성질이나 크기에 따라 정해지며, 일반적으로 온도 등의 변화에 대해 안정하고 Q값도 매우 높다.

이 성질을 이용해서 안정한 발진기나 필터를 만들 수 있다

2) 압전기 [壓電氣, piezoelectricity]

어떤 종류의 결정판(結晶板)에 일정한 방향에서 압력을 가하면 판의 양면에 외력(外力)에 비례하는 양·음의 전하(電荷)가 나타나는 현상.

피에조전기라고도 한다. 예를 들면 수정에서 결정판을 잘라내어 x축 또는 y축을 따라 압력을 가하면 판의 양면에는 각각 전하가 생기고, 장력(張力)을 가할 때에는 이것과 반대부호의 전하가 생긴다. 1880년에 프랑스의 J.퀴리와 P.퀴리 형제가 전기석(電氣石)에서 발견한 현상인데, 수정·전기석·로셀염 등 예로부터 알려져 있는 것 외에 티탄산바륨·인산이수소암모늄·타르타르산에틸렌디아민 등의 인공결정이 현저하게 압전성을 가지는 소자(素子)로서 개발되어 있다.

일반적으로 1장의 결정판에 의한 압전기는 극히 미약하나 금속박을 삽입하여 이것을 몇 장 겹치면 그 전기량은 충분히 측정할 수 있게 된다. 이것을 이용하면 기계적인 변형을 전기적으로 꺼낼 수 있으므로 마이크로폰이나 전축용 픽업 등에 오래전부터 이용되고 있으며, 이 경우에는 압전율이 큰 로셀염의 결정이 많이 사용되고 있다. 또, 압전성을 가지는 결정판에 고주파 전압을 걸면 판이 주기적으로 신축하며, 특히 전압의 주파수를 판의 고유진동수에 맞추면 공진(共振)하여 판이 강하게 진동한다. 이 현상을 역압전효과(逆壓電效果)라고 하며, 이것에 의해 강력하고 안정된 기계적인 진동이 얻어진다. 이것의 응용으로서는 휴대용 라디오에 쓰이는 결정수화기·결정스피커와 초음파의 음원(音源)으로서 사용되는 수정이나 티탄산바륨 등을 들 수 있다.

이 밖에도 압전기발진자(壓電氣發振子)나 압전기공진자(壓電氣共振子)가 있다. 이것은 결정판이 탄성체로서 치수 등에 의해 정해지는 일정한 고유 진동수를 가지며, 탄성진동과 전기진동이 압전율을 통해서 결합되는 결과로서 마치 대단히 예민한 공진점을 가진 공진회로와 같이 동작하는 것을 이용하는 것이다. 방송전파의 주파수 유지, 수정시계의 중추부 등에는 특정한 방위(方位)로 잘라낸 수정의 진동자가 사용되며, 또 압전기공진자는 선택도가 높은 결정여파기(結晶濾波器)로서 원거리 통신회로에 널리 쓰인다

2.7. 추진부

2.7.1. 서보모터 [servomotor]

: 서보모터란?

서보모터란 일반적인 DC모터와는 달리 펄스의 주기로 정,역, 정지의 동작을 하는 특수모터. 서보기구의 조작부로서 제어신호(制御信號)에 의해 부하(負荷)를 구동하는 장치.

서보모터의 동력원에 따라 전기식(서보 전동기), 공기식(공기 서보모터), 유압식(유압 모터) 등이 있으며, 보통 서보모터라고 하면 서보 전동기를 가리키는 경우가 많다.

서보 전동기는 빠른 응답과 넓은 속도제어의 범위를 가진 제어용 전동기로, 그 전원에 따라 직류 서보모터와 교류 서보모터로 분류된다. 교류 서보모터의 대부분은 3상 서보모터이다.

이것들은 정지·시동·역전 등의 동작을 반복하므로, 방열효과를 좋게 하거나, 동작의 변화가 빨라지도록 설계상 고려되어 있다. 서보모터는 제어신호에 따라 운전되며, 그러기 위해서는 제어신호를 받아 이것을 증폭하여 서보모터를 구동하는 장치가 필요하며, 이것을 서보증폭기라고 한다.

펄스폭으로 위치(각도)를 제어하는 부품으로

위치피드백을 내부에서 구성하고 있으므로 외부에서의 처리가 불필요함,

인가 펄스폭은 10~20ms주기로 펄스폭이 각도를 제어한다.

0.7ms 이하이면 0도 (좌측으로 90도)

1.5ms 에서는 90도 (중심)

2.3ms 이상에서는 180도 (우측으로 90도)

서보모터의 구조는 직류모터의 회전축에 감속기어를 연결하고

여러번 감속된 기어의 마지막 회전축을 외부로 인출하고 있다.

인출된 마지막 회전축의 내부에는 가변저항을 연결하여

이 회전축이 움직인 각도만큼 따라서 움직이는 가변저항의 저항값을

내장된 전자회로에 되먹여 입력된 펄스폭에 비례하는 각도를 출력한다.

서보모터는 작은 크기로 좌우 90도 이내의 각도를 움직이므로

커다란 감속비를 채용하여 상대적으로 큰 힘을 낼 수 있으며
토크는 2.5kg/cm 속도는 0.2~0.9초/60도 정도의 성능을 가진다.
또한 RC 비행기에 조립하기 위해 가변운 재질로 만들어져 있다.

서보모터는 일찌기 모형제품의 움직임 용으로 많이 이용되어 왔으나,
최근에는 로봇의 관절을 움직이게하는 구동장치로 사용하고 있다

2.7.2. 전동기 [電動機, electric motor]

전기에너지를 역학적에너지로 바꾸는 장치. 일반적으로 모터(motor)를 말한다.

전동기는 전기에너지를 이용하여 기계적 움직임(역학적에너지)을 만드는 장치이다. 반대로 발전기는 역학적에너지를 이용하여 전기를 생산하는 장치이다. 전동기와 발전기는 모두 에너지를 변환하는 장치로, 발전기는 전동기의 역할을 할 수 있고 반대로 전동기는 발전기로 작동할 수 있다. 거의 대부분의 전동기가 회전운동의 동력을 만들지만, 직선운동을 생산하기도 한다.

전동기의 분류

전원(電源)의 종류에 따라 직류전동기와 교류전동기로 분류되며, 교류전동기는 다시 3상교류용과 단상교류용으로 구분된다. 3상교류용은 1kW 정도 이상부터 수천 kW까지, 그리고 드물게는 1만 kW를 넘는 대형기가 있으며, 단상교류용은 수백 W 이하의 소형기에 사용한다.

전동기의 원리

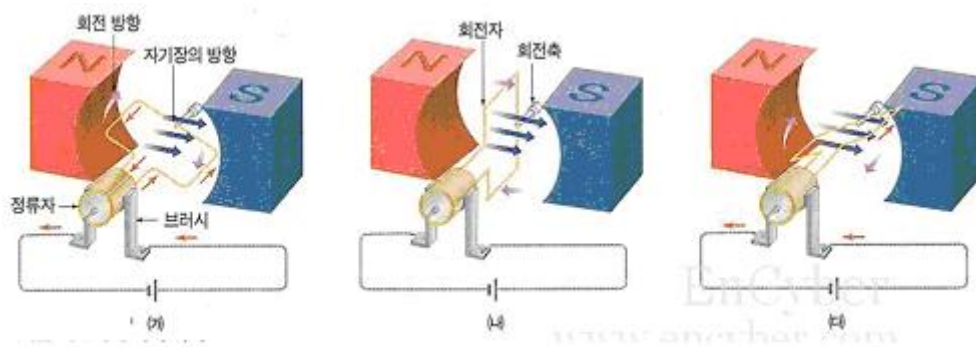


그림 2.18. 직류전동기의 회전원리

직류와 교류의 차이가 있어도 원리는 동일하다. 자기장 속에서 움직이는 전하는 로렌츠힘을 받는다. 따라서 전동기 내부에 영구자석을 놓아 자기장을 만들고, 축에 연결된 도선에 전류

를 흘리면 전자기력에 의해 회전한다. 이때 도선에 작용하는 전자기력은 자기장의 세기, 전류의 세기 및 도선의 길이에 비례한다

2.7.3. 필요한 개념

1) 로렌츠의 힘

전류가 흐르는 도선이 자기장 속에 있다면, 도선에 흐르는 전류는 자기장으로부터 힘을 받는다.

그러므로 전류는 전기를 띠고 있는 입자의 흐름이므로 자기장 속에서 운동하는 전하 역시 힘을 받아 운동하는 것이다.

아래 그림은 균일한 자기장 속에 전류가 흐르는 도선을 나타낸 것이다.

전류가 흐르는 방향과 도선 안에서의 자유 전자의 방향은 그림과 같이 반대의 방향이다.

전하량이 q 인 자유 전자가, v 의 속도를 가지고 t 초 동안에 균일한 자기장 B 에 수직으로 놓인, 길이 l 의 도선의 단면을 N 개 통과를 한다고 하자.

그러면, 이 도선을 흐르는 전류의 세기 I 는 단위 시간 동안에 통과한 자유 전자의 수와 같고, 자유 전자의 전하량이 q 이므로, 다음과 같이 나타낼 수 있다.

그러므로, 이 도선이 자기장 B 에 의해 받는 힘은 다음과 같다.

그러므로, 도선 내를 지나는 하나의 자유 전자가 받는 힘은 다음과 같이 나타낼 수 있다.

일반적으로 전하량 q 로 대전된 입자가 자기장 B 를 수직으로 v 의 속도로 가지고 입사하면, 이 입자가 받는 힘 F 는 $q\mathbf{v} \times \mathbf{B}$ 가 된다.

이 힘을 로렌츠의 힘이라고 한다.

여기서 힘의 방향은 플레밍의 왼손의 법칙으로 알 수 있다.

로렌츠의 힘은 입자가 자기장에 입사한 각도에 따라 다른 값을 가지게 된다.

아래 그림과 같이 전하량 q 로 대전된 입자가 자기장 B 를 v 의 속도로 각도 θ 로 입사하였다면, 이 입자가 받는 로렌츠 힘 F 는 다음과 같다.

그러므로 자기장의 방향과 입사한 입자의 방향이 나란하거나 정 반대가 되면 힘은 0이 되고, 자기장의 방향과 입자의 운동 방향이 직각이 되면 입자가 받는 힘은 최대가 된다.

옆의 그림은 대전된 입자가 균일한 자기장에 직각으로 입사된 그림을 나타낸 것이다.

이 입자의 질량을 m , 전하량을 q , 입사 속도가 v 라면, 이 입자는 자기장으로 부터 로렌츠의 힘을 받아 등속 원운동을 하게 된다.

이 때, 자기장을 B , 원 궤도의 반지름을 r 이라고 한다면 로렌츠의 힘이 바로 구심력의 역할을 하므로, 다음과 같은 식이 성립한다.

그러므로, 대전된 입자의 궤도 반지름 r 은 $r = \frac{mv}{qB}$ 가 된다.

대전된 입자가 원운동을 하는 주기 T 는 다음과 같이 나타낼 수 있다.

이것을 이용하여 만든 장치가 대전 입자의 가속장치인 사이클로트론이다.

만일 대전된 입자의 전하량이 같은 두 입자가 같은 속도로 입사를 하게 된다면, 반지름 r 은

대전 입자의 질량 m 에 비례하여 커진다.

이와 같은 원리로 만든 것이 바로 질량 분석기이다.

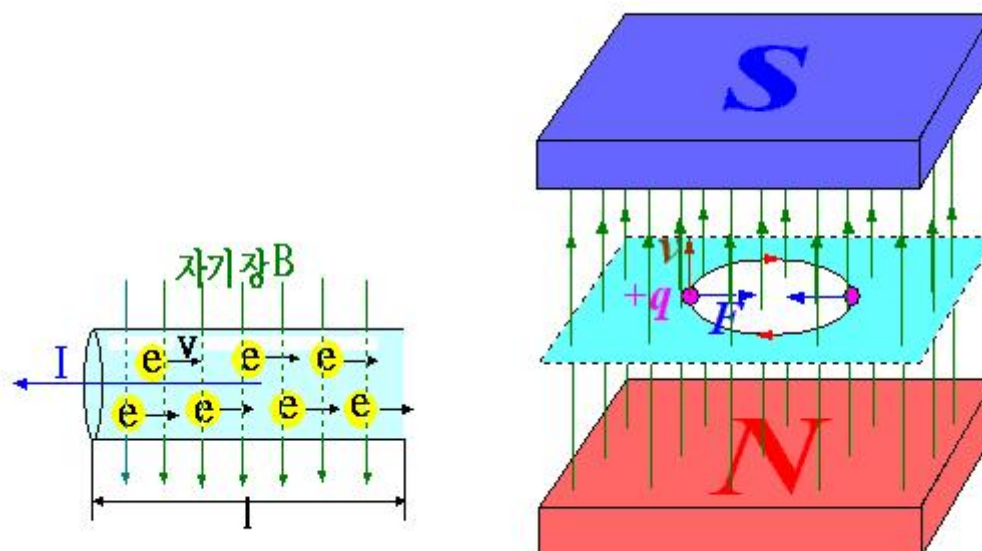


그림 2.19. 로렌츠의 힘

2) 플레밍의 왼손법칙

전류가 흐르고 있는 도선에 대해 자기장이 미치는 힘의 작용방향을 정하는 법칙이다. 전류가 흐르는 도선 하나하나의 부분이 자기장에 의해서 받는 힘은, 왼손의 중지를 전류가 흐르는 방향으로, 검지를 자기력선의 방향으로 향하게 하여, 이것들에 대해 수직으로 편 엄지가 가리키는 방향으로 작용한다. 단, 전류와 자기장의 방향이 평행일 때는 이와 같은 힘은 작용하지 않는다.

3) 변속기 [變速機, transmission]

각종 엔진에서 발생하는 동력을 속도에 따라 필요한 회전력으로 바꾸어 전달하는 변속장치. 트랜스미션이라고도 한다. 일반적인 자동차용 내연기관은 일정한 속도에서 토크가 최대가 되는 데 대하여 달리기 시작할 때에는 더 강한 토크와 낮은 회전을 필요로 하며, 속도가 빨라짐에 따라 토크보다도 회전속도가 필요하게 된다.

그래서 엔진의 회전을 일정하게 유지하기 위해 기어를 사용하여 출발할 때에는 회전속도를 줄임과 동시에 토크를 늘려 주고, 속도가 빨라짐에 따라 회전을 높이는 역할을 하는 것이 변속기이다.

3. 세부 제작 일정

10월

일	월	화	수	목	금	토
1	2	3	4	5	6	7
8	9	10	11 설계 발표	12 제작	13 및 제어기	14 소프트웨어
15 프로그래밍	16 작성	17	18 제작	19 및 제어기	20 소프트웨어	21 프로그래밍
22 작성	23	24 제작	25 및 제어기	26 소프트웨어	27 프로그래밍	28 작성
29 제작	30 및 제어기	31 소프트웨어				

11월

일	월	화	수	목	금	토
			1 프로그래밍	2 작성 및 시	3 제작마무리 운	4 및 전
5 발표준비	6 최종 발표	7	8	9	10	11 모형선박 성능 컨테스트
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30		

4. 역할 분담

선박 선형 설계	윤상훈 , 박한솔
제어 장치 설계 및 제작	손지호, 주경환, 김영민, 양해상
선박 제작 및 보고서 작성	전체 조원
파워포인트 및 발표	손지호

5. 후기

* 전체후기

제 2차 보고서가 완성 되었다. 이번에도 저번 보고서 때와 마찬가지로 잦은 시행착오와 도전을 거듭한 후, 힘들게 한 고비가 일단락 되었다.

먼저 선박 설계, 조교님들께서 이 작업을 하면서 자주 실패하고 다시 해야 하는 일이 발생할 거라고 말씀하신 것처럼 우리는 선박설계의 마무리 단계인 EZ-compart 단계에서 재화중량 미달로 처음부터 다시 시작하게 되었다. 밤새워서 반복하고 도전한 작업이 수포로 돌아갔을 때에는 정말 하늘이 무너지는 것 같았지만, 조원들 모두가 머리를 싸매고 다시 도전하여 좋은 결과를 얻게 되었다. 우리 모두가 팀 워크의 중요성을 새로이 깨달은 사건이었다.

두 번째로, 제어부를 제작하기 위해 수업시간에 들은 수업 내용과 (물론 전부 이해하는 것은 매우 힘들었다.) 각자 인터넷에서 습득한 내용을 더하여 필요한 지식을 얻을 수 있었다. 이번 보고서에서는 그 얻은 지식을 실제로 활용할 기회는 없었지만 실제 제작에 착수하면 많은 도움이 될 거라 생각한다.

전년도 일정보다 빠른 시기에 발표 일이 잡혀서 우리가 할 일은 선배들이 작년에 했던 것보다 다소 적은 듯한 느낌이 들은 것은 사실이다. 하지만 팀원모두가 그럼에도 불구하고 밤을 새가며 프로젝트를 진행시키기 위해 많은 노력을 했다. 우리가 만든 배가 멋지게 장애물을 피하며 물위를 달릴 그 때까지 우리 팀원들의 노력은 계속 될 것이다.

* 개인후기



손지호

첫 번째 프로포절에 이어서, 이번 중간 보고서에서는 실제 선박 설계 프로그램으로 직접 치수를 입력하여 선형을 설계하였다. 또한, 그로부터 배수량과 경하 중량을 추정하여 주어진 요구 조건인 재화 중량 6kg을 만족할 수 있는지 살펴 보았다.

그리고, 프로포절 때는 확실하게 알지 못하였던 제어부에 대해서 수업 내용을 바탕으로 조사하고 공부하면서, 이번 모형 선박 제작에 있어서 우리가 공부하고 실행해야 할 임무를 확실히 알 수 있었다.

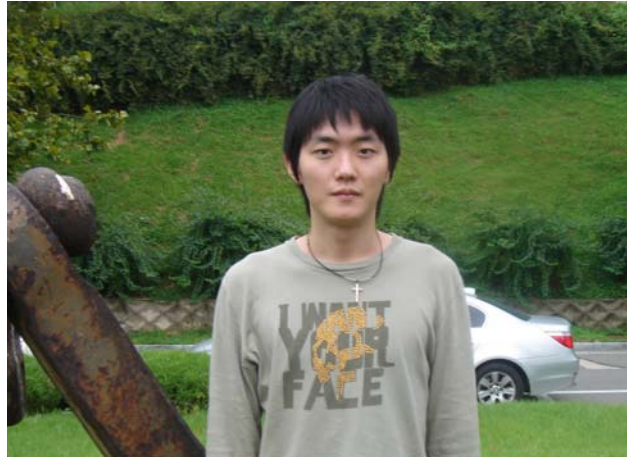
선형 설계의 경우, 한솔이와 상훈이가 도맡아서 추석 전부터 설계에 들어갔다. 시행 착오도 많았지만 둘이서 합심해서 조금씩 최종 선형을 완성해가는 모습을 보면서 자랑스럽고 든든했다.

처음에는 제어부에 대한 막연한 두려움에 선형 설계 부분에 조원 두 명만을 배치하여 솔직히 걱정도 많이 되었다. 그런데 별 문제 없이 설계를 진행하고 완성하는 모습을 보고서 제어부를 맡은 나머지 네 명의 조원은 더욱 막중한 책임감을 가지게 되었다.

제어부의 경우는, 교수님과 조교의 배려로, 앞으로도 과제 수행을 통해 전반적인 내용을 학습하게 될 것이다. 실제로 기관에 납땜을 통해 여러 전기 부품을 장착하고 연결해서 회로를 완성하고, PC 상에서 C언어로 프로그래밍을 하여, 기관 상의 메모리로 다운로드하여 자동으로 제어가 가능하도록 하는 것이 우리가 해야 할 일이다.

이번 보고서에 실제 제작에 들어가서 어느 정도 진척된 모습도 사진으로 올리려고 했으나, 긴 추석 연휴를 맞아 고향에 내려가 열심히 쉬는 바람에 실천에 옮기지 못하는 못하였다. 하지만 폭 쉬고 학교로 돌아온 만큼 앞으로 뻑뻑한 모형 선박 제작 과정에 있어서 힘차서 일정을 추진해 가고자 한다.

보고서 발표를 마치면 바로 제어부 제작을 시작하게 될 텐데, 전기 부품들을 다루는 작업에 나와 조원들은 많은 호기심을 느끼고 있으며, 보람있고 실속있는 경험이 될 것이라 믿고 있다. 앞으로 하게 될 작업에 대한 부담감도 있지만, 그 이상의 기대감을 가지고 있다. 조원들과 협력하여, 즐겁게 제작 일정을 추진하고 성공적으로 마무리 하는 것이 나의 바람이다.



김영민

이번 중간 보고서를 작성하면서 매우 고생한 것 같다. 초반 proposal 보고서를 준비할 때만 해도 주위에서 그리고 심지어 매스컴에서도 익히 들었던 이 조선해양공학계획 과목에 대해서 충분히 각오하고 있다고 생각했지만 점점 더 과정이 진행되면서 힘들다는 것을 알게 되었다. 선주의 요구에 따라 배를 만들어 간다는 것이 많은 유연성을 필요로 한다는 것을 알 수 있었다. 더불어 기초는 배웠으나, 선체를 제작하기 위한 실용적인 시작을 배움과 동시에 보고서를 작성하는 일은 더욱 순발력과 응용이 필요한 일이었다.

우리 조는 선체 부와 제어 부 파트로 사람을 나누어 했는데 나는 제어 부를 담당했다. 우선 선체부를 ezship을 통해서 선형을 설계하였는데, 그 과정이 순탄치 않았다. CP 커브를 구하는데 수많은 trial이 있었고 4차까지 진행된 밤샘 작업을 통해서 수정할 수 있었다. 매끄러운 선형이 나오기까지 계속 반복되는 수정을 거쳤고, 그렇게 얻어낸 선형이기에 더욱 애착을 가지게 되었다. 그렇게 만족하지 않고, 조금 더 괜찮은 선형을 위해 노력하는 조원들의 모습은 많은 것을 느끼게 해주었다.

그 모습에 자극 받아서 나도 제어파트에서 나의 맡은 바를 충실히 이행하려 애쓰게 되었다. 비록 전자 회로와 C 언어쪽에 전문적인 지식을 가지고 있진 않아도 최대한 이번 프로젝트 모형선에 들어가는 제어 파트쪽에 개념적인 설명과 부연적인 내용들을 최대한 조사하여 완성된 중간보고서를 위해서 많은 노력을 하였다. 어디선가 보기만 했던 부품들의 개념과 필요성을 찾아보면서 많은 것을 알고 느낄 수 있었다.

추석 때 조차 쉬지 않고 작업하는 조원들의 모습을 보면서 하나의 목표를 위해 같이 노력하고서 완성될 배를 생각하니 흐뭇하였다. 제어부분은 우리 과의 전공 분야가 아니기 때문에 힘든 점도 있었지만 하나하나씩 조사해 나가며 배워나가는 부분도 굉장히 유익하고 도움이 되었다. 아직은 직접 배를 제작하기 이전이지만, 이런 과정을 밟아가면서 완성될 배를 생각하며, 더욱 힘내고 더욱 열심히 제작에 전념해야겠다. 이 순간들이 분명 시간이 지난 미래에 내게 지탱해줄 수 있는 기억이 될 것이고, 필요한 지식이 될 것임을 이미 느끼고 있다. 마치 실제로 선주의 요구를 받은 것처럼, 실제 배를 만드는 것처럼, 그런 마음가짐으로 남은 작업 역시 최선을 다해 나아갈 것이다!



박한솔

Proposal에 이은 이번 보고서에는 설계부분을 맡아서 하게 되었다. 상훈이와 내가 설계 부분을 맡았는데 수업을 하면서 내용을 배워가면서 선형설계를 우선 시작하였다. 생각대로 작업은 매우 까다로웠다. 기준이 되는 점을 찍은 후 기준이 되는 선을 긋고 섹션라인을 그어서 선형을 완성하는 작업이다. 맨 처음에는 많은 시행착오를 거치면서 하나하나씩 완성을 해 나아갔다. 처음에 일단 선형을 완성한 후 방형계수를 맞추기 위해 조금씩 부피를 조정했다. 점을 하나하나씩 옮겨서 부피를 조정하는 작업이 여간 피곤한 것이 아니었다. 일단 선형을 완성한 후 추석이 다가왔는데, 어디선가 컴파트 부분은 금방 할 수 있다는 소리를 듣고 추석 때는 꼭 쉴 수 있겠구나 생각했다. 하지만 생각대로 꼭 쉬었지만 실상은 그렇지 않았다. 컴파트 부분도 상당히 시간이 걸렸다. 남은 시간이 얼마 되지 않아서 밤을 새어 가며 작업을 했다. 간신히 컴파트 부분을 끝낸 후에는 밤을 샌 피로도 풀 새 없이 바로 경화중량의 추정에 들어갔다. 경화중량의 추정이 이전까지 했던 것보다 더 번거로운 작업이었다. 부재를 하나하나 나눈 뒤 좌표를 찍고 길이를 구해 넓이를 구하는 과정이었다. 근데 더더욱 슬펐던 것은 경화중량까지 다 구하고 나니 재화중량이 6kg이 되지 않아 배를 다시 설계해야 했다는 사실이었다. 잠도 자지 못하고 너무 힘들었다. 하지만 여기서 포기하면 기한이 얼마 남지 않아 보고서를 쓸 수가 없었다. 그래서 설계부분 두 명에서 다시 세분하여 내가 보고서를 지금까지 한 것 만큼만 쓰기로 하고 상훈이는 설계를 다시 하기로 하여 새로운 마음으로 시작했다. 다시 시작하여 배수량을 늘리는데 성공하고, 재화중량까지 맞추어서 경화중량을 재 추정하고 보고서의 설계부분을 완성하는데 성공했다.

정말 정신 없이 달려왔다. 설새 없이 좌표를 생각하고, 계산하고 선을 긋고... 정말 정신없이 이틀이 지나갔지만 남는 것이 많은 것 같다. 우선 배의 기본적인 형상이 어떻게 설계되는지 직접 참여해 보았고 1학기 때 이론으로만 보았던 여러 가지 용어들을 되새김하는 계기가 될 수 있었다. 앞으로의 제작에도 열심히 참여하여 배를 성공적으로 만들겠다.



양해상

9월 20일 proposal 발표, 10월 11일 중간 보고서 발표, 어느새 3주가 지났다.

Proposal 제작한지가 엇그제 같은데, 이제 중간 보고서 제작을 마치니 시간이 무척 빨리 가는 것 같다.

정신 없이 지나간 시간 속에서, 우리가 얼마나 효율적으로, 그리고 확실하게 선체제작을 준비하고 있는 지가 매우 중요하다고 생각한다.

우선 선형설계부분은 -이것은 내 담당이 아니지만- EzShip 이라는 다루기 쉬운 프로그램을 이용하여 설계를 한 결과, 비교적 만족스러운 결과가 나왔다. 담당자인 -매끄러운 Cp Curve를 자랑스럽게 보여준- 박한솔군과 윤상훈군에게 매우 감사하게 생각한다.

다음은 내가 맡은 제어부분을 보면, 이제승 조교님의 쉽고 친절함 가르침 하에 -시키는 대로만 하면 되었지만- 중간 보고서 제작을 마칠 수 있었다.

이번 project에서 가장 주력해야 할 부분이 바로 제어부분이다. 맡은 책임이 막중한 만큼 무엇보다도 확실하게 준비하지 않으면 안 된다. 내가 특기할 만한 사항은 회로도 부분이다. 딱 봐도 복잡 • 난해해 보이는 회로를 작성하려니 처음부터 공부를 해야 했다. 기초적인 전자전기소자, 회로기호 등을 익히고, 마이크로 프로세서 • 마이크로 컨트롤러 등을 조사하면서 기본지식을 쌓았다. 그리하여 회로를 대략적으로 구상했고 마지막으로 회로가 복잡한 -그래서 회로 대신 사진을 넣었다- 센서는 공부를 더 해야 하는 부분이다.

제어부분은 우선 회로를 구상하고 빨리 제작에 들어가는 것이 최선이기 때문에 약간 늦은 감이 없지 않으므로 조원들 개개인의 노력이 절실히 필요할 것이다.

중간 발표 이후부터 본격적으로 제작에 들어간다. 멋지게 움직이는 우리의 배를 떠올리며 『Designare』 힘내자!!



윤상훈

이번 설계보고서에서 나는 선체부문을 맡았다. 사실 제어부는 너무 어려울 것 같아서 그나마 쉬워 보이는 선형설계를 한다고 하였다. 하지만 선형설계도 만만치 않았다. 계속해서 연습을 해야 잘 할 수 있는데 설계 프로그램은 과 전산실에 있는 몇몇 컴퓨터에만 깔려 있는 데다가 추석 연휴라 학교에 자주 오기도 힘든 상황이었다.

처음에는 막막하고 짜증만 났지만 선형이 하나씩 완성 되어가는 걸 보니 기분이 좋았다. 선형 설계가 비교적 쉬운 작업이라고 생각해서 빨리 마무리 짓고 어려운 제어부에 조원들과 같이 매달려 볼 계획이었는데 역시나 추석에 열심히 노느라 거의 하지 못했다. 결국 또 밤새워 하게 되었다. 이번 중간발표를 준비하면서 설계라는 것이 얼마나 어려운지 뼈저리게 느꼈다. 다시 하고 또 다시하고...

이번 중간 발표를 마치고 나면 부지런히 작업해서 빠른 시일 내로 선체를 완성할 수 있었으면 좋겠다. 지금 당장은 우리가 만들 배가 물에 잘 뜨기만 해 줬으면 하는 바람이다.



주경환

마침내 제 2차 중간 보고서를 마무리 했다. 나는 제어부를 맡았는데, 추석 연휴 동안 너무 열심히 쉬느라 내 할 일을 많이 해내지 못했다. 보고서를 막 쓸 때쯤 돼서 다른 조원들에 비해 제어부에 대한 사전지식이 부족하다는 것을 깨닫게 되었고, 부족한 지식을 보충하기 위해 부랴부랴 인터넷으로 관련 site를 돌아다니며 급하게 공부를 했다. 생소한 정보들에 처음에는 많이 낯설었지만, 계속 공부를 해나가다 보니 그것 나름대로의 재미가 있다는 것을 깨달았다. 특히 직접 기판을 사고 여러 가지 회로를 연결하는 것에 조원들 모두가 큰 관심을 갖고 있다. 중학교 기술 가정시간 이후로는 납땜을 해보지 못한 이유인지, 아니면 남자로서 직접 기계 만지는 일을 한번 해보고 싶다는 본능인지는 모르겠지만 선체보다는 모두들 직접 제작에 있어서 제어부 쪽에 더 관심을 갖고 있다. 또한 이렇게 직접 제작하는 것뿐만 아니라 C언어를 이용하여 프로그램을 짜는 것도 이제 슬슬 공부 준비를 하고 있다. 많은 시간과 노력을 필요로 하겠지만 새로운 분야에 대한 지식일수록 재미있다는 것을 알기에 많은 기대를 하고 있다. 그리고 ez 프로그램을 이용하여 설계작업을 하느라 자주 밤을 샌 한솔이와 상훈이에게 너무 고맙다. 열심히 일한 설계부에 지지 않도록 내가 속한 우리 제어부도 열심히 해야겠다는 생각을 갖게 되었다.

이제 본격적인 제작에 들어 가게 되었다. 지금까지와는 비교도 안될 정도로 많은 시행착오와 시련이 있을 것으로 예상된다. 하지만 우리 배가 수조 위에서 뜬 모습으로 장애물을 피해 목적지에 도달할 그 날까지 열심히 할 것이다.