Chapter-11 BAS Software

ဤအခန်းကိုလေ့လာပြီးနောက်

(က)　BAS software structure နှင့် BAS software hierarchy ကို နားလည်သဘောပေါက်စေရန်

(ခ)　Database တည်ဆောက်နည်း(development) အမျိုးမျိုးကို နားလည်သဘောပေါက်စေရန်

(ဂ)　DDC text based ၊ graphical programming နှင့် DDE သဘောတရား(concept) နားလည်သဘောပေါက်စေရန်

၁၁.၁ BAS Software Structure

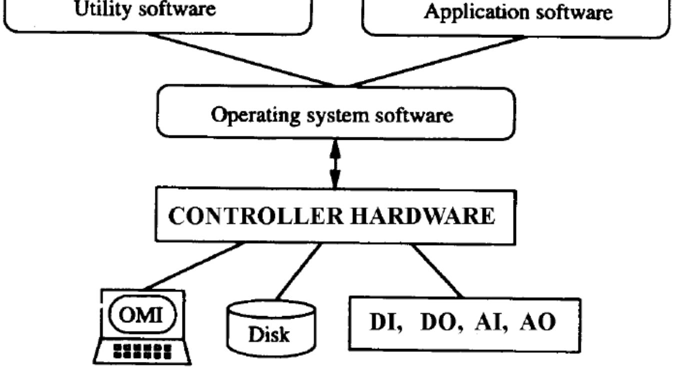
BAS နည်းပညာတွင် အမျိုးမျိုးသော software architecture များ ပုံစံအမျိုးမျိုးဖြင့် ရှိနိုင်သည်။ ကွန်ပြူတာနည်းပညာကို အခြေခံ၍ တည်ဆောက်ထားသည့် BAS network များတွင် Workstation PC ၊ communication unit ၊ DDC စသည်တို့ မလွဲမသွေ ပါဝင်ကြသည်။ BAS software architecture တစ်ခုတွင် အောက်ပါ software များ ပါဝင်ကြသည်။

(၁) Operating System(OS) software

(၂) Utility software နှင့်

(၃) Application software တို့ ဖြစ်သည်။

Software သုံးမျိုး(three classes) အပြန်အလှန် ဆက်သွယ်ထားပုံကို ပုံ(၁၁-၁)တွင် ဖော်ပြထားသည်။



ပုံ ၁၁-၁ Hierarchy of BAS software structure

၁၁.၁.၁ Operating System (Lowest Level)

Operating system software သည် memory location စီမံခန့်ခွဲပေးခြင်း၊ ရိုက်ထည့်သည့် စာလုံး(text)များကို လက်ခံပေးခြင်း စသည့် hardware-specific function များကို ဆောင်ရွက်ပေးသည်။ ကွန်ပြုတာတိုင်းတွင် Operating System(OS) software အနည်းဆုံးတစ်မျိုး ရှိရန် လိုအပ်သည်။ Windows 95၊ Windows NT၊ Windows XP ၊ Windows 7 စသည်တို့သည် Operating System(OS) software များဖြစ်ကြသည်။

၁၁.၁.၂ Utility Software

DDC အတွင်းရှိ software များသည် အလိုလျောက် အဆက်မပြတ် run နေကြသည်။ သို့သော် အချို့သော operation များသည် တစ်ခါတစ်ရံမှသာ operator ၏ ခိုင်းစေမှုအောက်တွင် (run)ရန် လိုအပ်သည်။ ထိုကဲ့သို့ တစ်ခါတစ်ရံမှသာ run ရန် လိုအပ်သည့် software ကို "**Utilities Software**" ဟုခေါ်သည်။

Communications functions ၊ database management functions ၊ operator-machine interface functions သို့မဟုတ် specific control algorithms စသည့် လုပ်ငန်းများကို utility software က ဆောင်ရွက်ပေးသည်။ Application program များမှ utility software modules ကို ခေါ်၍("called") တစ်ခုခု လုပ်ပေးရန် ခိုင်းစေနိုင်သည်။

၁၁.၁.၃ Application Program Software (Highest Level)

Application software များသည် BAS ၏ အဓိကလုပ်ငန်း(main function)များကို ဆောင်ရွက်ပေးသည့် program များ ဖြစ်ကြသည်။
အဓိကလုပ်ငန်း(main functions)များ အနက်မှ အချို့ကို အောက်တွင်ဖော်ပြထားသည်။

(၁) Data acquisition
(၂) Metering
(၃) Monitoring
(၄) Control
(၅) Energy management နှင့်
(၆) Operator-machine interface applications တို့ဖြစ်သည်။

Application program များသည် database များနှင့် အပြန်အလှန် စပ်ဆက်နေသည်။ BAS system တွင် တစ်ခုထက်ပိုများသည့် database များ ရှိနိုင်သည်။ Database များကို controlled plant နှင့် လုပ်ဆောင်ရမည့် specific function တို့၏ အချက်အလက်များပေါ်တွင် မူတည်၍ စနစ်တကျ အစီအစဉ်တကျတည်ဆောက်ထားသည်။

Communication network အတွင်း၌ အမျိုးမျိုးသော controller များ၊ communication node များ၊ စသည်တို့ အချက်အလက်များ ဖလှယ်ရန်အတွက် protocol တူညီရန် လိုအပ်သည်။ BAS application programme အမျိုးမျိုး ရရှိနိုင်သည်။ Application program များ၏ capability နှင့် effectiveness သည် control task အမျိုးအစား နှင့် DDC network ၏ architecture အပေါ်တွင်မူတည်သည်။

Typical BAS Application Software

Start/Stop Control

(က) Energy Management applications

(၁) Start/Stop Control
(၂) Duty Cycling
(၃) Load shedding
(၄) Temperature setback
(၅) Enthalpy control

(၆) VAV control
(၇) Optimal start/stop
(၈) Chiller and condenser optimization

(ခ) Energy Management applications

(၁) chilled water(cooling)
(၂) electricity
(၃) Gas
(၄) Water
(၅) fuel consumption(e.g. for standby generators)
(၆) advance warning and action required for maintenance service

(ဂ) Facilities Management

(၁) Fire detection, alarm and prevention
(၂) Access control and security
(၃) Intelligent Building applications
(၄) Operation and maintenance management (e.g. work order generation, maintenance procedures, management information)
(၅) Maintenance Scheduling, Inventory Control, Logistics
(၆) Office automation
(၇) Facility booking s

အမျိုးမျိုးသော application program များ တစ်ချိန်တည်းတွင် အတူ run နိုင်သည်။

၁၁.၂ BAS Software Design Concept

၁၁.၂.၁ DDC Software

DDC software များသည် DDC အား မည်ကဲ့သို့သော တွက်ချက်မှု ပြုလုပ်မည်၊ မည်သို့ဆောင်ရွက်ရမည်ဆိုသည့် ညွှန်ကြားချက်များ(set of instructions)နှင့် လုပ်ငန်းစဉ်များ(procedures) စီမံပေးမည်။ DDC software ကို DDC ၏ Random Access Memory (RAM) ပေါ်တွင် သိမ်းဆည်းထားလေ့ရှိသည်။ လိုအပ်သည့်အခါ operator မှ ပြင်ဆင်ခြင်း၊ ပြောင်းလဲခြင်း စသည်တို့ ပြုလုပ်နိုင်သည်။

၁၁.၂.၂ DDC Firmware

Firmware များသည် စက်ရုံမှ ထည့်ပေးလိုက်သည့် (factory-embedded) DDC software instruction များဖြစ်သည်။ DDC Read-Only Memory (ROM) ပေါ်တွင် အသေ(permanently)ရေးထားသည်။ System user မှ ပြင်ဆင်ပြောင်းလဲခြင်း ပြုလုပ်ခွင့် မရှိ။ Firmware programme ထဲတွင် DDC program module များဖြစ်သည့် on/off control ၊ PID control ၊ scheduling စသည်တို့ ပါဝင်သည်။

၁၁.၂.၃ DDC Database

DDC database တစ်ခုတွင် physical data point များအားလုံး ၏ အချက်အလက်များ ပါဝင်သည်။ DDC configuration လုပ်စဉ်အတွင်း database ကိုတည်ဆောက်ပြီး DDC memory (RAM) ပေါ်တွင် သိမ်းဆည်းထားသည်။

DDC database ၏ property များမှာ

(၁) RAM ပေါ်တွင် သိမ်းဆည်းထားသည့် DDC database ကို ပြင်ဆင်နိုင်သည်။ ပျက်(delete)နိုင်သည်။ အခြား memory device ဆီသို့ ရွှေ့ပြောင်း(move) နိုင်သည်။

(၂) DDC database object များသည် input/output အမျိုးအစားနှင့်သက်ဆိုင်သည့် ဒေတာများ(type of data) ၊ control programs ၊ variable တန်ဖိုးများ နှင့် constant တန်ဖိုးများ စသည်တို့ ဖြစ်သည်။

၁၁.၂.၄ DDC Database Configuration

DDC database configure ပြုလုပ်ခြင်းတွင် operational data နှင့် data point များ၏ အချက်အလက်များ DDC memory ပေါ်တွင် သိမ်းဆည်းရန် စနစ်တကျ ပြင်ဆင်ခြင်း ပါဝင်သည်။ BAS application programs များ executie လုပ်ရန်အတွက် database များ လိုအပ်သည်။ Database သည် workstation PC ၌ လည်းကောင်း အခြားသော central network communication unit များ၌ လည်းကောင်း ရှိနိုင်သည်။

DDC Software Modules

DDC software တွင် DDC ၏ memory (ROM) ပေါ်တွင် ရှိနေသည့် internal module များစွာ ပါဝင်သည်။ Software module များကို graphical နည်း သို့မဟုတ် text based နည်း(method)ဖြင့် configure လုပ်နိုင်သည်။

DDC software modules အမျိုးမျိုးကို ဖော်ပြထားသည်။

(၁) Control modules (e.g., on/off, PID)

(၂) Arithmetic modules (e.g. averaging, summing, totalization)

(၃) Programmable logic modules (PLC)

(၄) Interlock modules , etc (e.g., EF with EF OUT)

၁၁.၃ DDC Software Programming

ဆောင်ရွက်လိုသည့် control လုပ်ငန်းများကို program ရေး၍ DDC ၏ memory အတွင်းသို့ ထည့်ထား (download) ပေးရသည်။

နည်းနှစ်မျိုးဖြင့် program ရေးနိုင်သည်။

(၁) Text-based programming language

(က) Program များသည် DDC အား မည်ကဲ့သို့ လုပ်ဆောင်ရမည် အစီအစဉ်ကို ညွှန်ကြားသည်။

(ခ) အထူးကျွမ်းကျင်သူများသာ Program ရေးနိုင်သည်။

(၂) Graphical programming language(similar to that used for data base configuration)

(က) Program များသည် DDC အား မည်ကဲ့သို့ လုပ်ဆောင်ရမည်ကို ညွှန်ကြားသည်။

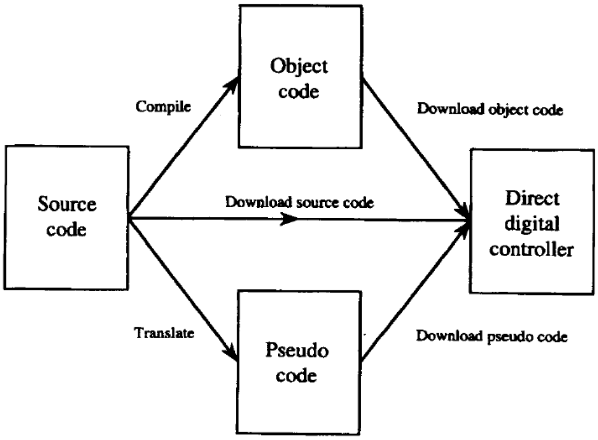
(ခ) Program ရေးသားတတ်ရန် လေ့ကျင့်ရာတွင် လွယ်ကူသည်။

(ဂ) အသုံးပြုသူများ လွယ်ကူ အဆင်ပြေသည်။ (user-friendly)

၁၁.၃.၁ Text-based programming

Translator

Text based programming language တွင် input text file ကို pseudo code ဟုခေါ်သည့် compressed လုပ်ထားသည့် binary file အဖြစ်သို့ ပြောင်းလဲသည်။ Compression လုပ်ထားသောကြောင့် DDC memory ပေါ်တွင် နေရာအနည်းငယ်သာယူသည်။ Control program အတိုင်း execute လုပ်ရန်အတွက် DDC operating system မှ pseudo code ၏ အဓိပ္ပာယ်ကို ဖွင့်ဆိုရသည်။



ပုံ ၁၁-၂ Text based programming language တစ်ခု၏ ဥပမာ

Compiler

Compiler သည် input text file ကို machine instruction များအဖြစ်သို့ ပြောင်းပေးသည်။

Compile လုပ်ထားသည့် machine instruction များကို DDC(microprocessor) သည် နားလည်နိုင်ပြီး control လုပ်ငန်းများဆောင်ရွက်ပေးသည်။

```
010   A=(TEMP;AI)-(PID;SP)
020   IF A<5 THEN 30 ELSE 80
030   IF A<-5 THEN 40 ELSE 80
040  (PID;PI)=0
050  (PID;PE)=0
060  (PID;RP)=(PID;DP)
070   GOTO 20
080   B=A*(PID;DP)/100
090   IF(PID;DP)>=100 THEN 120 ELSE 100
100   IF(PID;DP)<=0 THEN 120 ELSE 100
110  (PID;PI)=A+(PID;PE)/2*(PID;IC)+(PID;P1)
120   C=(PID;P1)/100
130  (PIC;PE)=A
140   A=(PIC;RP)+B+C
150   IF A>=0 THEN 170 ELSE 160
160   A=0
170   IF A<=100 THEN 190 ELSE 180
180   A=100
190   IF(SAFETY;BV)==1 THEN 200 ELSE 210
200   A=O
210  (PID;DP)=A
220   SWAIT 1
230   GOTO 10
```

ပုံ ၁၁-၃ Text-based programming

Text-based programming တစ်ခု၏ ဥပမာ

DDC program များကို ရေးသားပြီးနောက် translate လုပ်ခြင်း၊ compile လုပ်ခြင်း စသည်တို့ ကိစ္စများ ဆောက်ရွက်ပြီးနောက် DDC အတွင် memory ပေါ်သို့ download လုပ်ထည့်ရသည်။

Text-based method ၏ အားသာချက်များမှ

(၁) Parameter များ အားလုံးကို စာကြောင်းတစ်ကြောင်းတည်းဖြင့် ရေးသားနိုင်သည်။

(၂) စာလုံးများ(text) ဖြင့်ရေးသားထားသောကြောင့် တကူးတက မှတ်တမ်းတင်ရန် မလိုအပ်ပေ။ Entry

format များ၊ syntax များ ကို ကျွမ်းကျင်သူ အမှားရှာဖွေနိုင်သည်။

(၃) Program အတွင်း၌ မှတ်ချက်များ၊ comment များကို အတူတကွရေးသားနိုင်သောကြောင့် အခြားသူများ ဖတ်ရှုရာတွင် နားလည်သဘောပေါက်ရန် ပိုမို လွယ်ကူသည်။

(၄) ပြန်လည်စစ်ဆေးရန်(reviewing) အတွက် print ထုတ်နိုင်သည်။

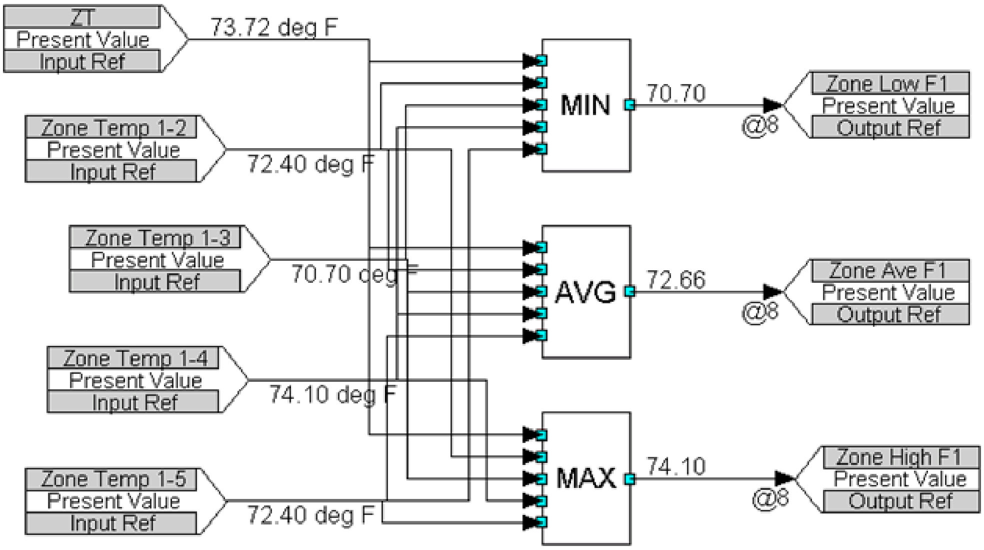
Text-based method ၏ အားနည်းချက်များမှာ

(၁) Syntax များကို အလွန်တိကျစွာ ရေးသားရန်လိုသည်။ database entry statement များကို သိရန် လိုအပ်သည်။

(၂) DDC ၏ memory ထဲသို့ download မလုပ်ခင် compile လုပ်ရန် အဆင့် တစ်ဆင့် ပိုလုပ်ရန်လိုသည်။

(၃) Compile လုပ်သည့် အဆင့်သို့ရောက်မှသာ အမှားများကို တွေ့နိုင်သည်။

၁၀.၃.၂ Programming Language (GPL)



ပုံ ၁၀-၄ Example of Graphic Programming Language(GPL)

အထက်ပါ ပုံ(၁၀-၄) Programming သည် ဇုံ(၅)ခု မှ Temperauter များ 1-1 မှ 1-5 ကို အနိမ့်ဆုံး တန်ဖိုး(minimum value)၊ အမြင့်ဆုံးတန်ဖိုး(maximum value) နှင့် ပျမ်းမျှတန်ဖိုး(average value)တွက်ပုံ ဖော်ပြသည့် program ဖြစ်သည်။ Graphics diagram တွင် ပါရှိသည့် MIN block ၊ MAX block ၊ AVG block တို့သည် software "objects" များဖြစ်ကြသည်။

MIN block သည် ဇုံ(Zone)(၅)ခု မှ temperauter များအနက်မှ အနိမ့်ဆုံး အပူချိန်ကို ရွေးချယ်ပေးသည်။
MAX block သည် ဇုံ(Zone)(၅)ခု မှ temperauter များအနက်မှ အမြင့်ဆုံး အပူချိန်ကို ရွေးချယ်ပေးသည်။
AVG block သည် ဇုံ(Zone)(၅)ခု မှ temperauter များ ပျမ်းမျှ အပူချိန်ကို တွက်ပေးသည်။

၁၀.၄ Logic Concept

DDC များ အလုပ်လုပ်ပုံ နှင့် အသုံးပြုနိုင်ပုံများကို နားလည်ရန်အတွက် logic concept ကို ပထမဦးစွာ နားလည်ရန် လိုအပ်သည်။ အခြေခံ logic function (၃)မျိုးဖြစ်သည့် AND ၊ OR နှင့် NOT တို့ကို ကွဲပြားအောင် electric circuit များ နှင့် ladder diagram များနှင့်တကွ အသေးစိတ် ရှင်းလင်း ဖော်ပြထားသည်။ ထို logic function (၃)မျိုးကို အခြေခံ၍ ရိုးရှင်းလွယ်ကူသည့် control decision များမှ စ၍ ခက်ခဲရှုပ်ထွေးသည့် control decision များအထိ တည်ဆောက်နိုင်သည်။

၁၁.၄.၁ Binary Concept

Binary concept သည် အိုင်ဒီယာအသစ်တစ်ခု မဟုတ်ပါ။ Binary concept ကို ရှေးယခင်ကတည်းက စတင်သုံးစွဲခဲ့ကြသည်။ ဥပမာ-မီးလုံး တစ်လုံးသည် ပွင့်(on) နေနိုင်သည်။ ပိတ်(off) နေနိုင်သည်။ ခလုပ်(switch) တစ်ခုသည် open ဖြစ်နေနိုင်သည် သို့မဟုတ် close ဖြစ်နေနိုင်သည်။ မော်တာတစ်လုံးသည် မောင်း(running)နေနိုင်သည်။ ရပ်(stop) နေနိုင်သည်။ Digital System များတွင် အခြေအနေ(၂)မျိုး(two state condition)သာ ရှိနိုင်သည်။ Signal ရှိခြင်း (signal activated) သို့မဟုတ် မရှိခြင်း(not activated)၊ high သို့မဟုတ် Low ၊ on သို့မဟုတ် off ၊ open သို့မဟုတ် close စသည့်တို့ ကဲ့သို့ အခြေအနေ(၂)မျိုးအနက်မှ တစ်ချိန်တွင် တစ်မျိုးသာ ဖြစ်နေနိုင်သည်။ ထို two state concept (binary concept)သည် decision များချရန် အတွက် အခြေခံဖြစ်သည်။ Binary concept သည် DDC များ၏ function block များ ပြုလုပ်ရန် နှင့် digital computer များ၏ အခြေခံ(fundamental) သဘောတရားများ ဖြစ်သည်။

PLC စာအုပ်များတွင် Binary 1 သည် signal ရှိခြင်းကို ဆိုလိုသည်။ Binary 0 သည် signal မရှိခြင်း (absence of signal)ကို ရည်ညွှန်းသည်။ Digital system များတွင် ဤအခြေအနေ(၂)မျိုး(two state)ကို မတူညီသည့် voltage level (၂)ခုဖြင့် ဖော်ပြသည်။ +V volt နှင့် 0V volt တို့ကို Table 11-1 တွင် ဖော်ပြထားသည်။ Volt တစ်ခုသည် တခြား Volt တစ်ခုထက် အပေါင်းဓါတ်အား သို့မဟုတ် အဖိုဓါတ်အား (more positive) ပိုများသည်။ တစ်ခါတစ်ရံ Binary 1 သို့မဟုတ် logic 1 ကို true ၊ on ၊ high စသည် အဓိပ္ပာယ်များ အဖြစ် ရည်ညွှန်းပြောဆိုပြီး binary 0 သို့မဟုတ် logic 0 ကို false ၊ off ၊ low စသည့် အဓိပ္ပာယ်များ အဖြစ် ရည်ညွှန်း ပြောဆိုသည်။

Table 11-1. Binary concept using positive logic.

Example	Logic 1 (+V)	Logic 0 (0V)
Limit switch	Operating	Not operating
Bell	Ringing	Not ringing
Light bulb	On	Off
Horn	Blowing	Silent
Motor	Running	Stopped
Clutch	Engaged	Disengaged
Valve	Closed	Open

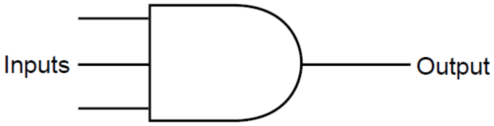
၁၁.၄.၂ Logic Function

Table 11-1 တွင်ဖော်ပြထားသည့် more positive voltage ကို logic 1 အဖြစ် သတ်မှတ်သည်။ Less positive voltage ကို Logic 0 အဖြစ် သတ်မှတ်ထားသည်။ မိမိနှစ်သက်သလို အဆင်ပြေသလို ရွေးချယ် သတ်မှတ်ထားခြင်း ဖြစ်သည်။

Binary concept သည် physical quantities (Binary variables) များ၏ အခြေအနေ(၂)မျိုး (two state)ကို 0 နှင့် 1 ဖြင့် ဖော်သည်။ အခြေအနေ(၂)မျိုးအနက် တစ်ချိန်၌ အခြေအနေတစ်မျိုးသာ ဖြစ်နိုင်သည်။ (၂)ခု ထက်ပိုသည့် binary variable များ ပေါင်းစပ်ပြီး ရရှိသည့်ရလာဒ်ကို True (1) သို့မဟုတ် False (0) အခြေအနေ တစ်မျိုးမျိုးဖြင့် ဖော်ပြပုံ ဆက်လက် လေ့လာမည်။ DDC များသည် logical statement များကို အခြေခံ၍ ဆုံးဖြတ်ချက်များ ချမှတ်(make decision) ကြသည်။ DDC များသည် digital equipment များ ဖြစ်ကြသောကြောင့် AND ၊ OR နှင့် NOT စသည့် အခြေခံ(fundamental) logic function (၃)မျိုးကို အခြေခံ၍ operation များ ပြုလုပ်ကြသည်။ ထို function များ ပေါင်းစပ်ပြီး binary variable များကို statement အဖြစ် တည်ဆောက်ကြသည်။

Function တိုင်းတွင် စည်းကမ်းချက်(rule)များ ရှိကြသည်။ ထိုစည်းကမ်းချက်(rule)များအရ ရလာဒ်(result)ကို ထုတ်ပေးသည်။ ဆုံးဖြတ်ပေးသည်။ ကိုယ်ပိုင်သင်္ကေတ(symbol)များ ရှိကြသည်။ နားလည်မှု ရှင်းလင်းစေရန်အတွက် ရလာဒ်(result of statement)ကို output ဟု သတ်မှတ် ခေါ်ဆိုသည်။ Output ကို အင်္ဂလိပ်အက္ခရာ(Y)ဖြင့် သတ်မှတ်ဖော်ပြသည်။ Statement များ၏ အခြေအနေ(condition)များကို input (A နှင့် B) များဟု သတ်မှတ်သည်။ Input နှင့် output (၂)ခုစလုံးကို အခြေအနေ(၂)မျိုး (two state variable) ဖြင့် ဖော်ပြနိုင်သည်။

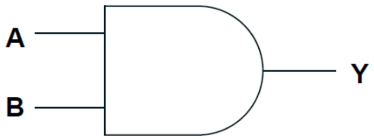
၁၁.၄.၃ "AND" Function



ပုံ ၁၁-၅ Symbol for the AND function.

ပုံ(၁၁-၅)သည် AND gate တစ်ခု၏ သင်္ကေတ(symbol) ဖြစ်သည်။ AND function ကို ပုံဖြင့် ဖော်ပြထားခြင်း ဖြစ်သည်။ Input များအားလုံး True (1) ဖြစ်မှသာ AND output သည် True (1) ဖြစ်သည်။

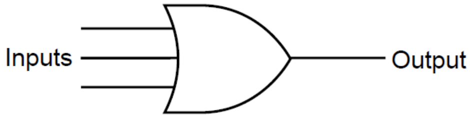
The AND output is True (1) only if all inputs are True (1).



AND Truth Table		
Inputs		Output
A	B	Y
0	0	0
0	1	0
1	0	0
1	1	1

ပုံ ၁၁-၆ Two-input AND gate and its truth table.

၁၁.၄.၄ "OR" Function



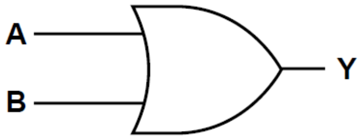
ပုံ ၁၁-၇ Symbol for the OR function.

ပုံ(၁၁-၇)သည် OR gate တစ်ခုကို သင်္ကေတ (Symbol) ဖြင့် ဖော်ပြထားသည်။ OR function ကို ပုံဖြင့် ဖော်ပြထားခြင်း ဖြစ်သည်။

The OR output is True (1) if on or more inputs are True (1).

OR gate ၏ input များအားလုံးအနက်မှ အနည်းဆုံး input တစ်ခု True ဖြစ်လျှင် output သည် True (1) ဖြစ်သည်။

AND function (AND Gate) နှင့် OR function (OR Gate) တို့တွင် input ပေါင်းများစွာရှိနိုင်သော်လည်း output တစ်ခုသာ ရှိမည်။ ပုံ(၁၁-၈)တွင် OR function ၏ true table ကို ဖော်ပြထားသည်။ input များ အားလုံး၏ ဖြစ်နိုင်သည့် combination ကို အခြေခံ၍ output Y ကို ထုတ်ပေးထားသည်။

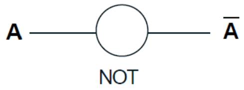


OR Truth Table		
Inputs		Output
A	B	Y
0	0	0
0	1	1
1	0	1
1	1	1

ပုံ ၁၁-၈ Two-input OR gate and its truth table.

၁၁.၄.၅ "NOT" function

ပုံ(၁၁-၉) NOT function ကို သင်္ကေတ (symbol) ဖြင့် ဖော်ပြထားသည်။ NOT function ၏ input မှာ FALSE (0) ဖြစ်လျှင် output သည်TRUE (1) ဖြစ်သည်။ အပြန်အလှန်အားဖြင့် input မှာ TRUE (1) ဖြစ်လျှင် output သည် FALSE (0) ဖြစ်သည်။ NOT Gate ၏ operation မှာ input ၏ state ကို ပြောင်းပြန်(inverse) လုပ်ပေးရန် ဖြစ်သည်။ ထို့ကြောင့် NOT Gate ကို inverter ဟုလည်း ခေါ်သည်။



NOT Truth Table	
Input	Output
A	$\bar{A}$
0	1
1	0

ပုံ ၁၁-၉ Symbol for the NOT function.

ပုံ ၁၁-၁၀ NOT gate and its truth table.

> အခြားသော AND function နှင့် OR function များနှင့် မတူသည့်အချက်မှာ NOT function တွင် input တစ်ခုတည်းသာ ပါရှိခြင်းဖြစ်သည်။ လက်ခံနိုင်ခြင်းဖြစ်သည်။ NOT function ကို တစ်ခုတည်း အသုံးပြုလေ့ မရှိပေ။ AND gate သို့မဟုတ် OR gate တို့ဖြင့် တွဲ၍ အသုံးပြုလေ့ရှိသည်။

ပုံ(၁၁-၁၀)တွင် NOT Operation နှင့် ယင်း၏ true table ကို ဖော်ပြထားသည်။ A အပေါ်တွင်ဘား (Bar) ကလေးတင်၍ $\bar{A}$ အဖြစ်ဖော်ပြခြင်းသည် NOT A ကို ရည်ညွှန်းသည်။

NOT function အလွယ်တကူသိမြင်နိုင်ရန်(Visualize လုပ်ရန်) ခဲယဉ်းသည်။ AND နှင့် OR တို့၏ function ကို သိမြင်နိုင်ရန်(visualize လုပ်ရန်) လွယ်သည်။ သိလွယ်၊ မြင်လွယ်သည်။ သို့သော် NOT function သည် အလွန်ရိုးရှင်းပြီး၊ အလွန်အသုံးဝင်ကြသည်။

အစပိုင်းတွင် ရှင်းပြထားသည့် အချက်(၃)ချက်

(၁) 1 သို့ 0 မိမိနှစ်သက်သည့်အတိုင်း ရွေးချယ်ပါ။

(၂) Logic 1 သည် ပုံမှန်အားဖြင့်(များသောအားဖြင့်) TRUE ၊ HIGH ၊ ON ဖြစ်သည်။ Logic 1 သည် Device တစ်မျိုးမျိုးကို Activate လုပ်ရန် ရည်ရွယ်သည်။ ဥပမာ- Output Y=1 ဖြစ်လျှင် မော်တာကို မောင်းရန် ဖြစ်သည်။

(၃) Logic 0 သည် ပုံမှန်အားဖြင့် (များသောအားဖြင့်) FALSE ၊ LOW ၊ OFF ဖြစ်သည်။ Logic 0 သည် device တစ်မျိုးမျိုးကို deactivate လုပ်ရန် ဖြစ်သည်။

ဥပမာ- Output Y သည် 0 ဖြစ်လျှင် မော်တာကို ရပ်ရန် ဖြစ်သည်။ အကယ်၍ ထိုအခြေအနေကို ပြောင်းလိုလျှင် NOT function ကိုသုံးသည်။ ဥပမာ- logic 0 သည် Device တစ်မျိုးမျိုး Active လုပ်လိုလျှင် NOT function ကို အသုံးပြုသည်။ ဥပမာ- output Y=0 ဖြစ်လျှင် မော်တာကိုမောင်းရန်ဖြစ်သည်။ NOT function သည် အသုံးဝင်သည်။

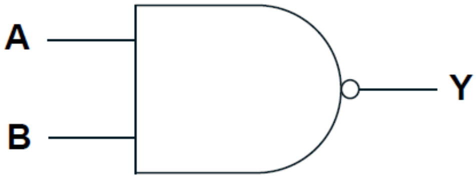
(က) NOT ကိုအသုံးပြုသည်။ Logic 0 (low condition) ဖြစ်လျှင် Device တစ်မျိုးမျိုးကို Active လုပ်ရန်

(ခ) Logic 1 (High Condition) ဖြစ်လျှင် Device တစ်မျိုးမျိုးကို deactivate လုပ်ရန်အတွက် NOT ကိုအသုံးပြုသည်။

အောက်တွင် NOT function ကို သုံးထားသည့် ဥပမာ(၂)ခုကို ဖော်ပြထားသည်။

NOT function ကို AND နှင့် OR function များဖြင့်အသုံးပြုလေ့ရှိကြသော်လည်း ပထမဥပမာတွင် NOT function တစ်ခုတည်းကို သာသုံးထားသည်။

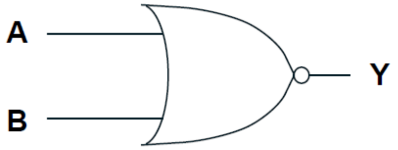
၁၁.၄.၆ NAND Gate



NAND Truth Table		
Inputs		Output
A	B	Y
0	0	1
0	1	1
1	0	1
1	1	0

ပုံ ၁၁-၁၁ Two-input NAND gate and its truth table.

ပုံ 3-7 တွင် NAND gate ၏ logic symbol နှင့် truth table ကိုဖော်ပြထားသည်။ OR gate ၏ output ၌ NOT သင်္ကေတ (symbol)ထည့်လျှင်လည်း အလုပ်လုပ်ပုံမှာ တူညီသည်။ OR gate ၏ normal output ကို function ထည့်ထားသောကြောင့် NOR gate ဖြစ်သည်။ ပုံ(၁၁-၁၂) တွင် NOR gate ၏ logic symbol နှင့် truth table ကိုဖော်ပြထားသည်။



NOR Truth Table		
Inputs		Output
A	B	Y
0	0	1
0	1	0
1	0	0
1	1	0

ပုံ ၁၁-၁၂ ပုံ **3-8.** Two-input NOR gate and its truth table.

၁၁.၅ Review Questions

(၁) Explain briefly about the software architecture applied to BAS.

(၂) List three classes of software used in building automation systems, and outline the functions of each type.

(၃) Describe three typical application software made available to BAS. Give examples on their real-life applications.

(၄) (က) (What is OOP?) Explain the term "object oriented programming" as applied to

building automation systems.

(ခ) Give one example of a DDC database entry that conforms with OOP, ignoring the actual syntax used.

(၅) Describe the information contained in a DDC database.

(၆) Explain the difference between "software" and "firmware", stating clearly, where each one resides in a computer system.

(၇) (က) Explain the purpose of a DDC database.

(ခ) Describe two methods commonly used for DDC database configuration in BAS industry.

(ဂ) Explain the difference between DDC database configuration and DDC programming.

(၈) (က) Describe two methods used to develop BAS application programs.

(ခ) Give two advantages and disadvantages for each method.

(၉) Explain the terms compiler and translator as used in computer programming.

(၁၀) Give two advantages of graphical programming method over test-based method in the development of BAS / DDC application programming.

(၁၁) List and describe five types of DDC program modules used for building control applications.

(၁၂) List three BAS application programs typically available/used for:-

(က) energy management

(ခ) metering and monitoring of building services

(ဂ) facilities management

(၁၃) Describe, with the aid of diagrams, the following real-time energy management routines generally used in BAS:-

(က) Time and temperature profiling

(ခ) Optimal start-stop

(ဂ) Duty cycling

(ဃ) Time and calendar scheduling

(၁၄) The facility management has to deploy the efficient energy management routines to their car parks in three basements of a commercial building. Propose some useful "energy management routines" which will be suitable for basement car park lighting and ventilation systems. Explain clearly how to implement this proposal with your programming knowledge.

-End-

Chapter-11 BAS Software

၁၁.၁ BAS Software Structure 1
၁.၁.၁ Operating System (Lowest Level) 2
၁၁.၁.၂ Utility Software 2
၁၁.၁.၃ Application Program Software (Highest Level) 2
၁၁.၂ BAS Software Design Concept 3
၁၁.၂.၁ DDC Software 3
၁၁.၂.၂ DDC Firmware 3
၁၁.၂.၃ DDC Database 3
၁၁.၂.၄ DDC Database Configuration 4
၁၁.၃ DDC Software Programming 4
၁၁.၃.၁ Text-based programming 4
၁၁.၃.၂ Programming Language (GPL) 6
၁၁.၄ Logic Concept 6
၁၁.၄.၁ Binary Concept 7
၁၁.၄.၂ Logic Function 7
၁၁.၄.၃ "AND" Function 8
၁၁.၄.၄ "OR" Function 8
၁၁.၄.၅ "NOT" function 9
၁၁.၄.၆ NAND Gate 10
၁၁.၅ Review Questions:- 10

Contents

၁၁.၁ BAS Software Structure 1
၁.၁.၁ Operating System (Lowest Level)..
၁၁.၁.၂ Utility Software ....
၁၁.၁.၃ Application Program Software (Highest Level)...
၁၁.၂ BAS Software Design Concept 3
၁၁.၂.၁ DDC Software...
၁၁.၂.၂ DDC Firmware ....
၁၁.၂.၃ DDC Database ....
၁၁.၂.၄ DDC Database Configuration..
၁၁.၃ DDC Software Programming 4

၁၁.၃.၁ Text-based programming..

၁၁.၃.၂ Programming Language (GPL) ....

၁၁.၄ Logic Concept...6

၁၁.၄.၁ Binary Concept..

၁၁.၄.၂ Logic Function ....

၁၁.၄.၃ “AND” Function ....

၁၁.၄.၄ “OR” Function ....

၁၁.၄.၅ “NOT” function..

၁၁.၄.၆ NAND Gate ....

၁၁.၅ Review Questions 10