

CHAPTER 4

Architecture of specific PLC controller

KJE555 - PLC

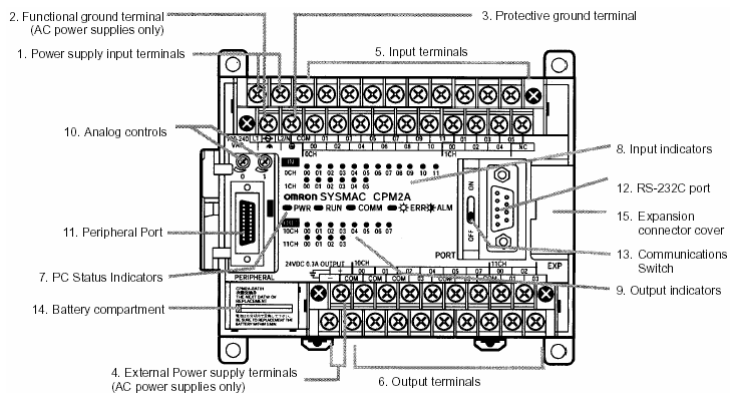
CPM2A PLC controller

- Each PLC is basically a microcontroller system (CPU of PLC controller is based on one of the microcontrollers, and in more recent times on one of the PC processors) with peripherals that can be digital inputs, digital outputs or relays as in our case

CPM2A PLC controller (2)

- Software itself is entirely different from assemblers used thus far, such as BASIC or C
- This specialized software is called "ladder"
- Specific look of CPM2A PLC controller can be seen in the following picture

CPM2A PLC controller (3)



CPM2A PLC controller (4)

- On the upper surface, there are 4 LED indicators (PWR, RUN, COMM, ERR) and a connection port with an RS232 module which is interface to a PC computer
- Aside from this, screw terminals and light indicators of activity of each input or output are visible on upper and lower sides

CPM2A PLC controller (5)

- Screw terminals serve to manually connect to a real system
- Hookups L1 and L2 serve as supply which is 220V~ in this case
- PLC controllers that work on power grid voltage usually have a source of direct supply of 24 VDC for supplying sensors and such (with a CPM2A source of direct supply is found on the bottom left hand side and is represented with two screw terminals)

CPM2A PLC controller (6)

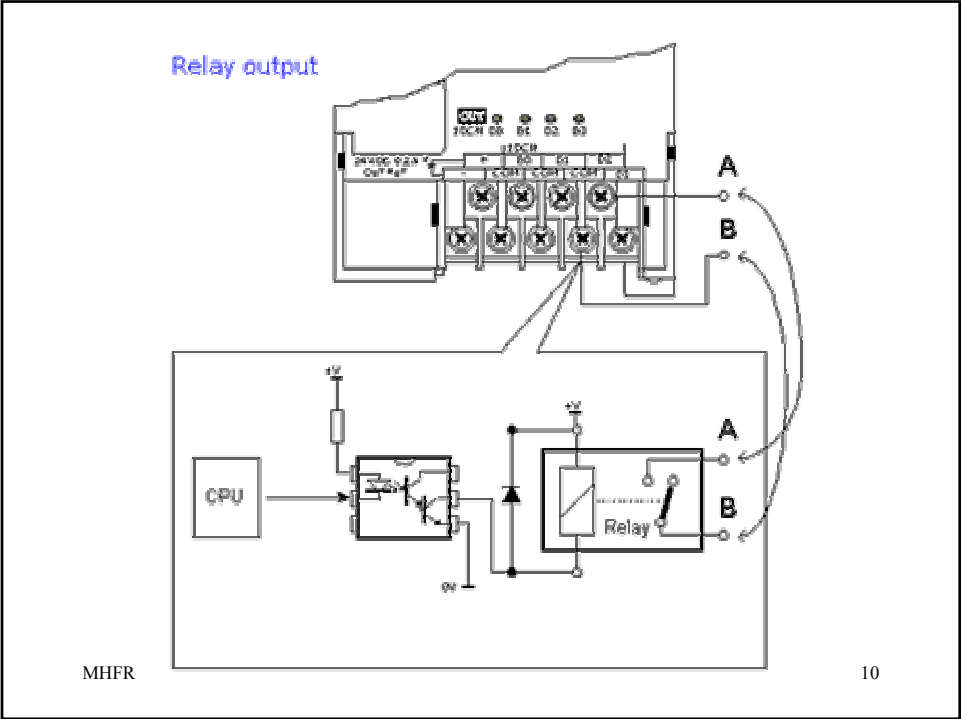
- To better inform programmers on PLC controller status, maker has provided for four light indicators in the form of LED's
- Beside these indicators, there are status indicators for each individual input and output
- If input/output is active, diode is lit and vice versa

PLC controller output lines

- Aside from transistor outputs in PNP and NPN connections, PLC can also have relays as outputs
- Existence of relays as outputs makes it easier to connect with external devices
- Model CPM2A contains exactly these relays as outputs

PLC controller output lines (2)

- There are 8 relays whose functional contacts are taken out on a PLC controller housing in the form of screw terminals
- In reality this looks as in the following picture (CPM1A)



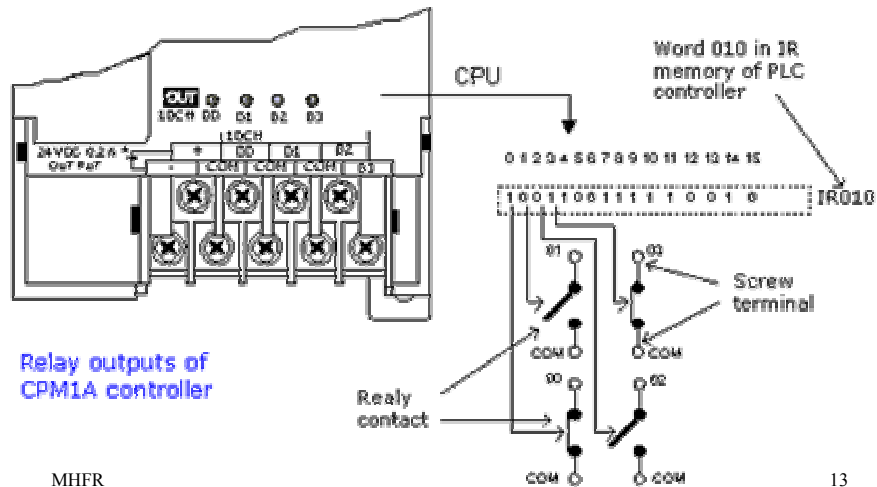
PLC controller output lines (4)

- With activation of phototransistor, relay comes under voltage and activates a contact between points A and B
- Contacts A and B can in our case be either in connection or interrupted
- What state these contacts are in is determined by a CPU through appropriate bits in memory location IR010

PLC controller output lines (5)

- One example of relay status is shown in a picture below ([CPM1A](#))
- A true state of devices attached to these relays is displayed

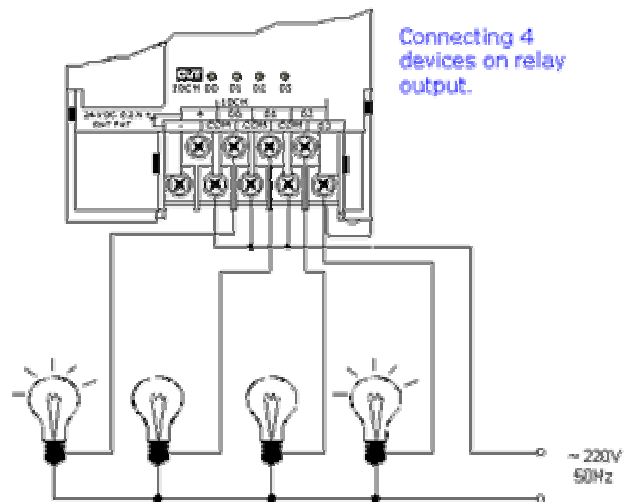
PLC controller output lines (6)



MHFR

13

PLC controller output lines (7)



MHFR

14

PLC controller input lines

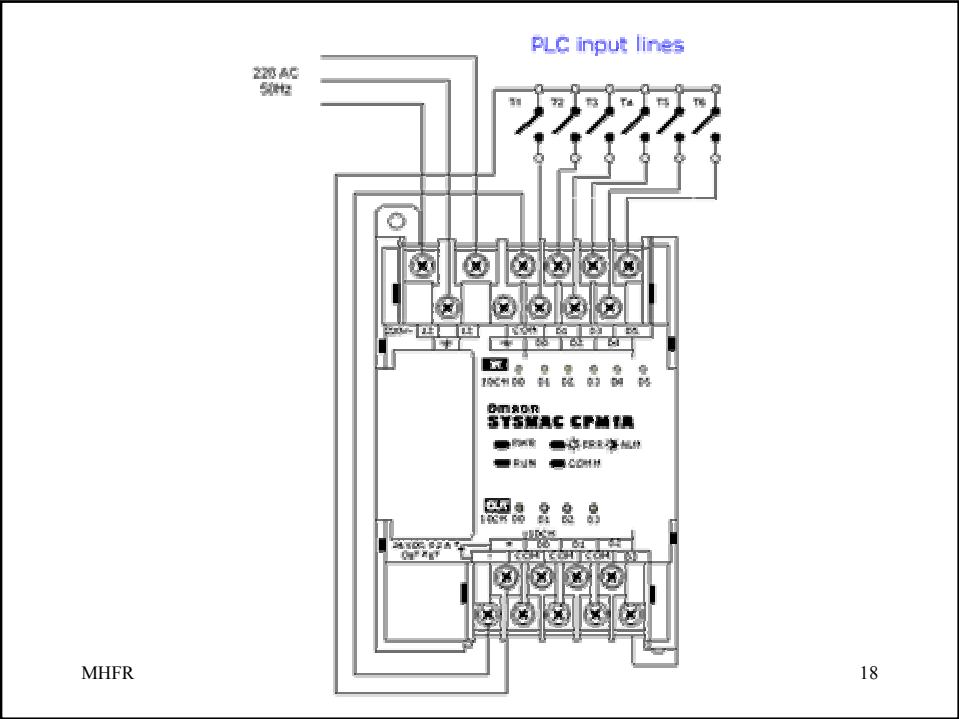
- Different sensors, keys, switches and other elements that can change status of a joined bit at PLC input can be hooked up to the PLC controller inputs
- In order to realize a change, we need a voltage source to incite an input

PLC controller input lines (2)

- The simplest possible input would be a common key
- As CPM2A PLC has a source of direct voltage of 24V, the same source can be used to incite input
- Problem with this source is its maximum current which it can give continually and which in our case amounts to 0.2A

PLC controller input lines (3)

- Since inputs to a PLC are not big consumers (unlike some sensor where a stronger external supply must be used) it is possible to take advantage of the existing source of direct supply to incite all twelve keys



How PLC controller works

- Basis of a PLC function is continual scanning of a program
- Under scanning we mean running through all conditions within a guaranteed period
- Scanning process has three basic steps:

How PLC controller works (2)

- *Step 1* - Testing input status
 - A PLC checks each of the inputs with intention to see which one of them has status ON or OFF. In other words, it checks whether a sensor, or a switch etc. connected with an input is activated or not. Information that processor thus obtains through this step is stored in memory in order to be used in the following step

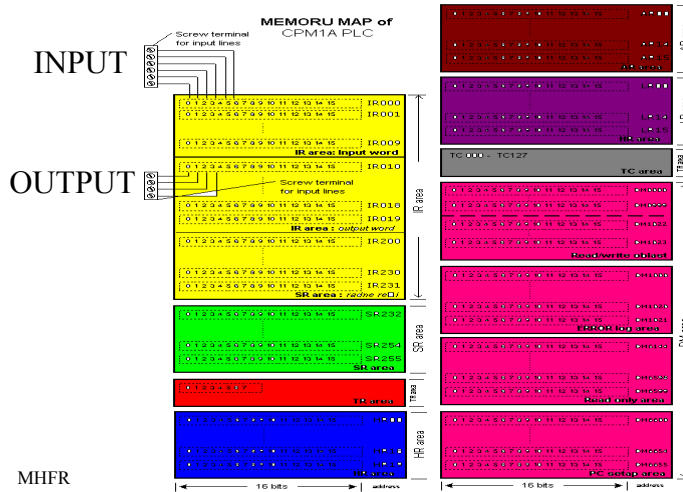
How PLC controller works (3)

- *Step 2 - Program execution*
 - A PLC executes a program, instruction by instruction. Based on a program and based on the status of that input as obtained in the preceding step, an appropriate action is taken. This reaction can be defined as activation of a certain output, or results can be put off and stored in memory to be retrieved later in the following step

How PLC controller works (4)

- *Step 3 - Checkup and correction of output status*
 - Finally, a PLC checks up output status and adjusts it as needed. Change is performed based on the input status that had been read during the first step, and based on the results of program execution in step two. Following the execution of step 3 PLC returns to the beginning of this cycle and continually repeats these steps. Scanning time is defined by the time needed to perform these three steps, and sometimes it is an important program feature

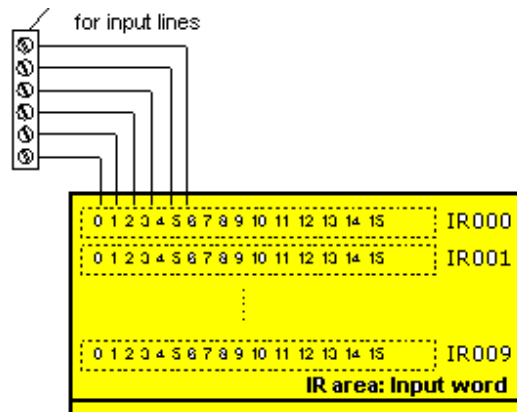
PLC controller memory map CPM1A



23

PLC controller memory map CPM1A (2)

- Input word

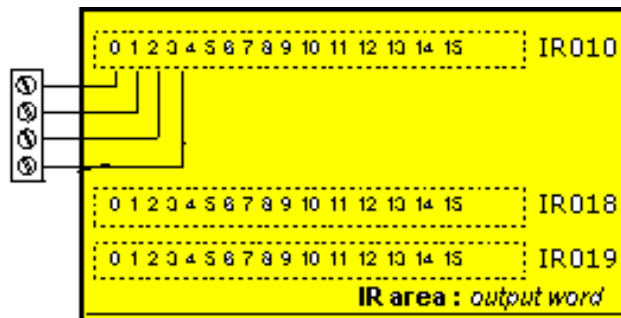


MHFR

24

PLC controller memory map CPM1A (3)

- Output word



MHFR

25

PLC controller memory map CPM1A (4)

- “201.7=1” would clearly indicate a word 201 and its bit 7 which is set to one
- IR region - Memory locations intended for PLC input and output
- SR region - Special memory region for control bits and flags

MHFR

26

PLC controller memory map CPM1A (5)

- TR region - When you move to a subprogram during program execution, all relevant data is stored in this region up to the return from a subprogram

PLC controller memory map CPM1A (6)

- HR region - It is of great importance to keep certain information even when supply stops. This part of the memory is battery supported, so even when supply has stopped it will keep all data found therein before supply stopped

PLC controller memory map CPM1A (7)

- AR region - This is one more region with control bits and flags. This region contains information on PLC status, errors, system time, and the like. Like HR region, this one is also battery supported
- LR region - In case of connection with another PLC, this region is used for exchange of data

PLC controller memory map CPM1A (8)

- Timer and counter region - This region contains timer and counter values. There are 128 values. Since we will consider examples with timers and counters, we will discuss this region more later on

PLC controller memory map CPM1A (9)

- DM region - Contains data related to setting up communication with a PC computer, and data on errors

Timers and counters

- There are two types of timers delay-off and delay-on
- First is late with turn off and the other runs late in turning on in relation to a signal that activated timers
- Example of a delay-off timer would be staircase lighting. Following its activation, it simply turns off after few minutes

Timers and counters (2)

- Each timer has a time basis, or more precisely has several timer basis
- Typical values are: 1 second, 0.1 second, and 0.01 second
- If programmer has entered 0.1 as time basis and 50 as a number for delay increase, timer will have a delay of 5 seconds ($50 \times 0.1 \text{ second} = 5 \text{ seconds}$)

Timers and counters (3)

- Typically, timers have two inputs
- First is timer enable, or conditional input (when this input is activated, timer will start counting)
- Second input is a reset input. This input has to be in OFF status in order for a timer to be active, or the whole function would be repeated over again

Timers and counters (4)

- With a PLC controller by Omron there are two types of timers: TIM and TIMH
- TIM timer measures in increments of 0.1 seconds
- It can measure from 0 to 999.9 seconds with precision of 0.1 seconds more or less

Timers and counters (5)

- Quick timer (TIMH) measures in increments of 0.01 seconds
- Both timers are "delay-on" timers
- They require assignment of a timer number and a set value (SV)
- When SV runs out, timer output turns on

Timers and counters (6)

- Numbers of a timing counter refer to specific address in memory and must not be duplicated (same number can not be used for a timer and a counter)

END OF CHAPTER 4