Chapter 1 \( \rightarrow \) Mfg, example of a PCB

Analog \( \rightarrow \) examples \( \rightarrow \) Radio, power, voltage, osc

Digital \( \rightarrow \) representation of pictures, 0/1, mproc

Analog \( \rightarrow \) continuous voltage (power/voltage, light, sound)

Digital \( \rightarrow \) discrete, 0/1, 0V & Vcc

on/off

---

Input - Digital

Port

4

\( 0 \) \( \text{OV or Vcc} \)

Input or Output:

Port direction

\( \text{Registers:} \)

\( \text{Data} \)

\( \text{Control} \)

\( \text{Example} = \) what is on \( \text{st}_-\text{port4} \)?
ECGR 4101/5101 Lecture 2

Data Port

B7 B6 B5 B4 B3 B2 B1 B0

1 0 1 0 1 0 1 0

Int myinput;

myinput = (int) s1_port4; (myinput is 170 decimal)

Port direction is controlled with another register

Actually in RX62N

PORTD.DDR.BYTE = 0x00; // inputs

Port D, bit 0

Data direction register

PORTD.DDR.BIT.B0 = 1; // direction = output
PORTD.DR.BIT.B0 = 1; // light the LED

Data Register
ECGR 4101/5101

Actual address of PORTD, PORTD.DDR is in an include file.

- PORTD = 0x0000 100E
- PORTD = 0x0000 103E

Include file have a mapping from ports (data registers) to hardware addresses.

Write the code:
1) Direction
2) Read the bit & put into the variable my_input
Answer:
```c
int myinput;
PORT4.DDR.BIT.B30 = 1;
myinput = (int) PORT4.DR.BIT.B30;
```

Write the code to:
1) Directions of everything
2) Read the button, and if pressed, light the LED
3) Do this Forever

Answer
```c
while (1) {
    // Hint: this means do forever
}
```
Answer

PORTA.DDR.BIT.BO = 0;  // push button -> input
PORTD.DDR.BIT.BO = 1;  // LED -> output

while (1) {
  if (!PORTA.DDR.BIT.BO) PORTD.DDR.BIT.BO = 1;
  else PORTD.DDR.BIT.BO = 0;

  // code...
}