

# Basic IO Interfacing on Firebird-V

e-Yantra Team  
Embedded Real-Time Systems Lab  
Indian Institute of Technology-Bombay

IIT Bombay  
October 10, 2014



# Agenda for Discussion

- 1 Input-Output Ports in ATmega2560
  - Overview of Ports
  - Accessing Ports
  - Examples
  
- 2 Write Your First Embedded C Program
  - Buzzer Interfacing
  - Programming Tools
  - C-code



# What are Ports?



# What are Ports?

- 1 Junctions where peripheral devices are connected.



# What are Ports?

- ① Junctions where peripheral devices are connected.
- ② Peripheral devices can be



# What are Ports?

- 1 Junctions where peripheral devices are connected.
- 2 Peripheral devices can be

## 1 Input Device

Example: Switch, Sensors, etc...



# What are Ports?

- ① Junctions where peripheral devices are connected.
- ② Peripheral devices can be

- ① Input Device

Example: Switch, Sensors, etc...

- ② Output Device

Example: Buzzer,LCD,Motors,LED, etc...



# PORTS in ATmega 2560





# PORTS in ATmega 2560

- 1 ATmega2560 is 100 pin controller



# PORTS in ATmega 2560

- 1 ATmega2560 is 100 pin controller
- 2 86 pins can be used as Input/Output Pins



# PORTS in ATmega 2560

- 1 ATmega2560 is 100 pin controller
- 2 86 pins can be used as Input/Output Pins
- 3 Pins are grouped together and called as PORT



# PORTS in ATmega 2560

- 1 ATmega2560 is 100 pin controller
- 2 86 pins can be used as Input/Output Pins
- 3 Pins are grouped together and called as PORT

1 ATmega2560 has Ten 8-bit ports

PORT<sub>x</sub>;                      x = A-to-F and H,J,K,L



# PORTS in ATmega 2560

- 1 ATmega2560 is 100 pin controller
- 2 86 pins can be used as Input/Output Pins
- 3 Pins are grouped together and called as PORT

- 1 ATmega2560 has Ten 8-bit ports

PORT<sub>x</sub>;                      x = A-to-F and H,J,K,L

- 2 ATmega2560 has One 6-bit port

PORTG



# PORTS in ATmega 2560

- 1 ATmega2560 is 100 pin controller
- 2 86 pins can be used as Input/Output Pins
- 3 Pins are grouped together and called as PORT

- 1 ATmega2560 has Ten 8-bit ports

PORT<sub>x</sub>;                      x = A-to-F and H,J,K,L

- 2 ATmega2560 has One 6-bit port

PORTG

- 4 All Port pins can be configured individually as input/output.



# Accessing PORTS



# Accessing PORTS

- ① Each Ports has three associated registers with it.





# Accessing PORTS

① Each Ports has three associated registers with it.

①  $\text{DDRx}$        $x = \text{A-to-H and J,K,L}$



# Accessing PORTS

① Each Ports has three associated registers with it.

① DDR<sub>x</sub>                      x = A-to-H and J,K,L

② PORT<sub>x</sub>                    x = A-to-H and J,K,L



# Accessing PORTS

① Each Ports has three associated registers with it.

① DDR<sub>x</sub>                      x = A-to-H and J,K,L

② PORT<sub>x</sub>                    x = A-to-H and J,K,L

③ PIN<sub>x</sub>                      x = A-to-H and J,K,L



# Understanding DDRx Register



# Understanding DDRx Register

## ① Data Direction Register



# Understanding DDRx Register

- ① Data Direction Register
- ② Purpose: To set PORT Pins as Input or Output



# Understanding DDRx Register

- ① Data Direction Register
- ② Purpose: To set PORT Pins as Input or Output
  - a.  $DDRx=0$  : PORTx is defined as INPUT



# Understanding DDRx Register

- ① Data Direction Register
- ② Purpose: To set PORT Pins as Input or Output
  - a.  $\text{DDR}_x=0$  :  $\text{PORT}_x$  is defined as INPUT
  - b.  $\text{DDR}_x=1$  :  $\text{PORT}_x$  is defined as OUTPUT





# Understanding DDRx Register

- ❶ Data Direction Register
- ❷ Purpose: To set PORT Pins as Input or Output
  - a.  $DDRx=0$  : PORTx is defined as INPUT
  - b.  $DDRx=1$  : PORTx is defined as OUTPUT
- ❸ Example: For PortB make lower nibble as input and upper nibble as output



# Understanding DDRx Register

- ❶ Data Direction Register
- ❷ Purpose: To set PORT Pins as Input or Output
  - a.  $DDRx=0$  : PORTx is defined as INPUT
  - b.  $DDRx=1$  : PORTx is defined as OUTPUT
- ❸ Example: For PortB make lower nibble as input and upper nibble as output



# Understanding DDRx Register

- ❶ Data Direction Register
- ❷ Purpose: To set PORT Pins as Input or Output
  - a.  $\text{DDR}_x=0$  :  $\text{PORT}_x$  is defined as INPUT
  - b.  $\text{DDR}_x=1$  :  $\text{PORT}_x$  is defined as OUTPUT
- ❸ Example: For PortB make lower nibble as input and upper nibble as output

D7	D6	D5	D4	D3	D2	D1	D0
1	1	1	1	0	0	0	0



# Understanding DDRx Register

- ❶ Data Direction Register
- ❷ Purpose: To set PORT Pins as Input or Output
  - a.  $\text{DDR}_x=0$  :  $\text{PORT}_x$  is defined as INPUT
  - b.  $\text{DDR}_x=1$  :  $\text{PORT}_x$  is defined as OUTPUT
- ❸ Example: For PortB make lower nibble as input and upper nibble as output

D7	D6	D5	D4	D3	D2	D1	D0
1	1	1	1	0	0	0	0

$\text{DDRB} = 0xF0$



# Understanding PINx Register



# Understanding PINx Register

- 1 Purpose: To Read data present on PORTx



# Understanding PINx Register

- 1 Purpose: To Read data present on PORTx
- 2 Save value of register in a variable



# Understanding PINx Register

- 1 Purpose: To Read data present on PORTx
- 2 Save value of register in a variable
- 3 Example:

Read Data from PORTC





# Understanding PINx Register

- 1 Purpose: To Read data present on PORTx
- 2 Save value of register in a variable
- 3 Example:

Read Data from PORTC



# Understanding PINx Register

- 1 Purpose: To Read data present on PORTx
- 2 Save value of register in a variable
- 3 Example:

Read Data from PORTC

PORTC =

D7	D6	D5	D4	D3	D2	D1	D0
1	1	1	1	0	0	0	0



# Understanding PINx Register

- 1 Purpose: To Read data present on PORTx
- 2 Save value of register in a variable
- 3 Example:

Read Data from PORTC

PORTC =

D7	D6	D5	D4	D3	D2	D1	D0
1	1	1	1	0	0	0	0

x= PINC



# Understanding PINx Register

- 1 Purpose: To Read data present on PORTx
- 2 Save value of register in a variable
- 3 Example:

Read Data from PORTC

PORTC =

D7	D6	D5	D4	D3	D2	D1	D0
1	1	1	1	0	0	0	0

x= PINC

x=0xF0



# Understanding PORTx Register



# Understanding PORTx Register

Case:1 When portx is defined as Output



# Understanding PORTx Register

Case:1 When portx is defined as Output

- ① Purpose: Output data present on PORTx pin



# Understanding PORTx Register

Case:1 When portx is defined as Output

- 1 Purpose: Output data present on PORTx pin
- 2 Example:





# Understanding PORTx Register

Case:1 When portx is defined as Output

- 1 Purpose: Output data present on PORTx pin
- 2 Example:



# Understanding PORTx Register

Case:1 When portx is defined as Output

❶ Purpose: Output data present on PORTx pin

❷ Example:

DDRA =

D7	D6	D5	D4	D3	D2	D1	D0
1	1	1	1	1	1	1	1



# Understanding PORTx Register

Case:1 When portx is defined as Output

- 1 Purpose: Output data present on PORTx pin
- 2 Example:

DDRA =

D7	D6	D5	D4	D3	D2	D1	D0
1	1	1	1	1	1	1	1

DDRA = 0xFF



# Understanding PORTx Register

Case:1 When portx is defined as Output

① Purpose: Output data present on PORTx pin

② Example:

DDRA =

D7	D6	D5	D4	D3	D2	D1	D0
1	1	1	1	1	1	1	1

DDRA = 0xFF

PORTA = 0xFF



# Understanding PORTx Register



# Understanding PORTx Register

case:2 When portx is defined as Input



# Understanding PORTx Register

case:2 When portx is defined as Input

- 1 Purpose: Activate/deactivate pull-up resistor



# Understanding PORTx Register

case:2 When portx is defined as Input

① Purpose: Activate/deactivate pull-up resistor

a.  $PORTx=1$  : Activate Pull-Up





# Understanding PORTx Register

case:2 When portx is defined as Input

① Purpose: Activate/deactivate pull-up resistor

a.  $PORTx=1$  : Activate Pull-Up

b.  $PORTx=0$  : Deactivate Pull-Up



# Understanding PORTx Register

case:2 When portx is defined as Input

① Purpose: Activate/deactivate pull-up resistor

a.  $PORTx=1$  : Activate Pull-Up

b.  $PORTx=0$  : Deactivate Pull-Up

② Example:



# Understanding PORTx Register

case:2 When portx is defined as Input

① Purpose: Activate/deactivate pull-up resistor

a.  $PORTx=1$  : Activate Pull-Up

b.  $PORTx=0$  : Deactivate Pull-Up

② Example:



# Understanding PORTx Register

case:2 When portx is defined as Input

❶ Purpose: Activate/deactivate pull-up resistor

a.  $PORTx=1$  : Activate Pull-Up

b.  $PORTx=0$  : Deactivate Pull-Up

❷ Example:

DDRA =

D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	0	0	0	0	0



# Understanding PORTx Register

case:2 When portx is defined as Input

① Purpose: Activate/deactivate pull-up resistor

a.  $PORTx=1$  : Activate Pull-Up

b.  $PORTx=0$  : Deactivate Pull-Up

② Example:

DDRA =

D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	0	0	0	0	0

$DDRA = 0x00$



# Understanding PORTx Register

case:2 When portx is defined as Input

① Purpose: Activate/deactivate pull-up resistor

a.  $PORTx=1$  : Activate Pull-Up

b.  $PORTx=0$  : Deactivate Pull-Up

② Example:

DDRA =

D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	0	0	0	0	0

$DDRA = 0x00$

$PORTA = 0xFF$



# Understanding PORTx Register

case:2 When portx is defined as Input

① Purpose: Activate/deactivate pull-up resistor

a.  $PORTx=1$  : Activate Pull-Up

b.  $PORTx=0$  : Deactivate Pull-Up

② Example:

DDRA =

D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	0	0	0	0	0

$DDRA = 0x00$

$PORTA = 0xFF$

Pull-Up is activated for Pins of PORTA.



# Examples

- ① Example 1: Make PORTD as output port and send bit pattern of D5





# Examples

- 1 Example 1: Make PORTD as output port and send bit pattern of D5



# Examples

- ❶ Example 1: Make PORTD as output port and send bit pattern of D5
- ❷ Step 1: Make Port D as Output port



# Examples

- 1 Example 1: Make PORTD as output port and send bit pattern of D5
- 2 Step 1: Make Port D as Output port



# Examples

- 1 Example 1: Make PORTD as output port and send bit pattern of D5
- 2 Step 1: Make Port D as Output port

DDRD =



# Examples

- 1 Example 1: Make PORTD as output port and send bit pattern of D5
- 2 Step 1: Make Port D as Output port

DDRD =

D7	D6	D5	D4	D3	D2	D1	D0
1	1	1	1	1	1	1	1



# Examples

- 1 Example 1: Make PORTD as output port and send bit pattern of D5
- 2 Step 1: Make Port D as Output port

DDRD =

D7	D6	D5	D4	D3	D2	D1	D0
1	1	1	1	1	1	1	1

DDRD = 0xFF



# Examples

- ❶ Example 1: Make PORTD as output port and send bit pattern of D5
- ❷ Step 1: Make Port D as Output port

$$\text{DDRD} =$$

D7	D6	D5	D4	D3	D2	D1	D0
1	1	1	1	1	1	1	1

$\text{DDRD} = 0xFF$

- ❸ Step 2: Put data on the Port D



# Examples

- 1 Example 1: Make PORTD as output port and send bit pattern of D5
- 2 Step 1: Make Port D as Output port

DDRD =

D7	D6	D5	D4	D3	D2	D1	D0
1	1	1	1	1	1	1	1

DDRD = 0xFF

- 3 Step 2: Put data on the Port D





# Examples

- 1 Example 1: Make PORTD as output port and send bit pattern of D5
- 2 Step 1: Make Port D as Output port

DDRD =

D7	D6	D5	D4	D3	D2	D1	D0
1	1	1	1	1	1	1	1

DDRD = 0xFF

- 3 Step 2: Put data on the Port D

PORTD =



# Examples

- ❶ Example 1: Make PORTD as output port and send bit pattern of D5
- ❷ Step 1: Make Port D as Output port

$$\text{DDRD} =$$

D7	D6	D5	D4	D3	D2	D1	D0
1	1	1	1	1	1	1	1

$\text{DDRD} = 0xFF$

- ❸ Step 2: Put data on the Port D

$$\text{PORTD} =$$

D7	D6	D5	D4	D3	D2	D1	D0
1	1	0	1	0	1	0	1



# Examples

- ❶ Example 1: Make PORTD as output port and send bit pattern of D5
- ❷ Step 1: Make Port D as Output port

$$\text{DDRD} =$$

D7	D6	D5	D4	D3	D2	D1	D0
1	1	1	1	1	1	1	1

$\text{DDRD} = 0\text{xFF}$

- ❸ Step 2: Put data on the Port D

$$\text{PORTD} =$$

D7	D6	D5	D4	D3	D2	D1	D0
1	1	0	1	0	1	0	1

$\text{PORTD} = 0\text{xD5}$



## Examples (Cont..)

- ① Example 2: Make PORTA input port with pull-up activated on all pin



## Examples (Cont..)

- 1 Example 2: Make PORTA input port with pull-up activated on all pin



## Examples (Cont..)

- 1 Example 2: Make PORTA input port with pull-up activated on all pin
- 2 Step 1: Make Port A as Input port



## Examples (Cont..)

- 1 Example 2: Make PORTA input port with pull-up activated on all pin
- 2 Step 1: Make Port A as Input port



## Examples (Cont..)

- 1 Example 2: Make PORTA input port with pull-up activated on all pin
- 2 Step 1: Make Port A as Input port

DDRA =





## Examples (Cont..)

- 1 Example 2: Make PORTA input port with pull-up activated on all pin
- 2 Step 1: Make Port A as Input port

DDRA =

D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	0	0	0	0	0



## Examples (Cont..)

- 1 Example 2: Make PORTA input port with pull-up activated on all pin
- 2 Step 1: Make Port A as Input port

DDRA =

D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	0	0	0	0	0

DDRA = 0x00



## Examples (Cont..)

- 1 Example 2: Make PORTA input port with pull-up activated on all pin
- 2 Step 1: Make Port A as Input port

DDRA =

D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	0	0	0	0	0

DDRA = 0x00

- 3 Step 2: To activate Pull-up Resistor send data on Port A



## Examples (Cont..)

- 1 Example 2: Make PORTA input port with pull-up activated on all pin
- 2 Step 1: Make Port A as Input port

DDRA =

D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	0	0	0	0	0

DDRA = 0x00

- 3 Step 2: To activate Pull-up Resistor send data on Port A



## Examples (Cont..)

- 1 Example 2: Make PORTA input port with pull-up activated on all pin
- 2 Step 1: Make Port A as Input port

DDRA =

D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	0	0	0	0	0

DDRA = 0x00

- 3 Step 2: To activate Pull-up Resistor send data on Port A

PORTA =



## Examples (Cont..)

- 1 Example 2: Make PORTA input port with pull-up activated on all pin
- 2 Step 1: Make Port A as Input port

DDRA =

D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	0	0	0	0	0

DDRA = 0x00

- 3 Step 2: To activate Pull-up Resistor send data on Port A

PORTA =

D7	D6	D5	D4	D3	D2	D1	D0
1	1	1	1	1	1	1	1



## Examples (Cont..)

- 1 Example 2: Make PORTA input port with pull-up activated on all pin
- 2 Step 1: Make Port A as Input port

DDRA =

D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	0	0	0	0	0

DDRA = 0x00

- 3 Step 2: To activate Pull-up Resistor send data on Port A

PORTA =

D7	D6	D5	D4	D3	D2	D1	D0
1	1	1	1	1	1	1	1

PORTA = 0xFF



## Examples (Cont..)

- ① Example 3: Make upper nibble of PORTD as output and lower nibble as input. Out 3 on upper nibble and read back data using PIND





## Examples (Cont..)

- 1 Example 3: Make upper nibble of PORTD as output and lower nibble as input. Out 3 on upper nibble and read back data using PIND



## Examples (Cont..)

- ❶ Example 3: Make upper nibble of PORTD as output and lower nibble as input. Out 3 on upper nibble and read back data using PIND
- ❷ Step 1: Make Port Ds Pin7 to Pin4 as output pins and Pin3 to Pin0 as input pins



## Examples (Cont..)

- 1 Example 3: Make upper nibble of PORTD as output and lower nibble as input. Out 3 on upper nibble and read back data using PIND
- 2 Step 1: Make Port Ds Pin7 to Pin4 as output pins and Pin3 to Pin0 as input pins



## Examples (Cont..)

- 1 Example 3: Make upper nibble of PORTD as output and lower nibble as input. Out 3 on upper nibble and read back data using PIND
- 2 Step 1: Make Port Ds Pin7 to Pin4 as output pins and Pin3 to Pin0 as input pins

DDRD =



## Examples (Cont..)

- 1 Example 3: Make upper nibble of PORTD as output and lower nibble as input. Out 3 on upper nibble and read back data using PIND
- 2 Step 1: Make Port Ds Pin7 to Pin4 as output pins and Pin3 to Pin0 as input pins

DDRD =

D7	D6	D5	D4	D3	D2	D1	D0
1	1	1	1	0	0	0	0



## Examples (Cont..)

- ❶ Example 3: Make upper nibble of PORTD as output and lower nibble as input. Out 3 on upper nibble and read back data using PIND
- ❷ Step 1: Make Port Ds Pin7 to Pin4 as output pins and Pin3 to Pin0 as input pins

DDRD =

D7	D6	D5	D4	D3	D2	D1	D0
1	1	1	1	0	0	0	0

DDRD = 0xF0



## Examples (Cont..)

- 1 Example 3: Make upper nibble of PORTD as output and lower nibble as input. Out 3 on upper nibble and read back data using PIND
- 2 Step 1: Make Port Ds Pin7 to Pin4 as output pins and Pin3 to Pin0 as input pins

DDRD =

D7	D6	D5	D4	D3	D2	D1	D0
1	1	1	1	0	0	0	0

DDRD = 0xF0

- 3 Step 2: Put data on Port Ds Pin7 to Pin4 and enable pullup for Pin3 to Pin0



## Examples (Cont..)

- 1 Example 3: Make upper nibble of PORTD as output and lower nibble as input. Out 3 on upper nibble and read back data using PIND
- 2 Step 1: Make Port Ds Pin7 to Pin4 as output pins and Pin3 to Pin0 as input pins

DDRD =

D7	D6	D5	D4	D3	D2	D1	D0
1	1	1	1	0	0	0	0

DDRD = 0xF0

- 3 Step 2: Put data on Port Ds Pin7 to Pin4 and enable pullup for Pin3 to Pin0





## Examples (Cont..)

- 1 Example 3: Make upper nibble of PORTD as output and lower nibble as input. Out 3 on upper nibble and read back data using PIND
- 2 Step 1: Make Port Ds Pin7 to Pin4 as output pins and Pin3 to Pin0 as input pins

DDRD =

D7	D6	D5	D4	D3	D2	D1	D0
1	1	1	1	0	0	0	0

DDRD = 0xF0

- 3 Step 2: Put data on Port Ds Pin7 to Pin4 and enable pullup for Pin3 to Pin0

PORTD =



## Examples (Cont..)

- ❶ Example 3: Make upper nibble of PORTD as output and lower nibble as input. Out 3 on upper nibble and read back data using PIND
- ❷ Step 1: Make Port Ds Pin7 to Pin4 as output pins and Pin3 to Pin0 as input pins

DDRD =

D7	D6	D5	D4	D3	D2	D1	D0
1	1	1	1	0	0	0	0

DDRD = 0xF0

- ❸ Step 2: Put data on Port Ds Pin7 to Pin4 and enable pullup for Pin3 to Pin0

PORTD =

D7	D6	D5	D4	D3	D2	D1	D0
0	0	1	1	1	1	1	1



## Examples (Cont..)

- ❶ Example 3: Make upper nibble of PORTD as output and lower nibble as input. Out 3 on upper nibble and read back data using PIND
- ❷ Step 1: Make Port Ds Pin7 to Pin4 as output pins and Pin3 to Pin0 as input pins

DDRD =

D7	D6	D5	D4	D3	D2	D1	D0
1	1	1	1	0	0	0	0

DDRD = 0xF0

- ❸ Step 2: Put data on Port Ds Pin7 to Pin4 and enable pullup for Pin3 to Pin0

PORTD =

D7	D6	D5	D4	D3	D2	D1	D0
0	0	1	1	1	1	1	1

PORTD = 0x3F



# Examples (Cont..)

- ❶ Example 3: Make upper nibble of PORTD as output and lower nibble as input. Out 3 on upper nibble and read back data using PIND
- ❷ Step 1: Make Port Ds Pin7 to Pin4 as output pins and Pin3 to Pin0 as input pins

$$\text{DDRD} =$$

D7	D6	D5	D4	D3	D2	D1	D0
1	1	1	1	0	0	0	0

$\text{DDRD} = 0xF0$

- ❸ Step 2: Put data on Port Ds Pin7 to Pin4 and enable pullup for Pin3 to Pin0

$$\text{PORTD} =$$

D7	D6	D5	D4	D3	D2	D1	D0
0	0	1	1	1	1	1	1

$\text{PORTD} = 0x3F$

- ❹ Step 3: Read the data from Port D



# Examples (Cont..)

- ❶ Example 3: Make upper nibble of PORTD as output and lower nibble as input. Out 3 on upper nibble and read back data using PIND
- ❷ Step 1: Make Port Ds Pin7 to Pin4 as output pins and Pin3 to Pin0 as input pins

DDRD =

D7	D6	D5	D4	D3	D2	D1	D0
1	1	1	1	0	0	0	0

DDRD = 0xF0

- ❸ Step 2: Put data on Port Ds Pin7 to Pin4 and enable pullup for Pin3 to Pin0

PORTD =

D7	D6	D5	D4	D3	D2	D1	D0
0	0	1	1	1	1	1	1

PORTD = 0x3F

- ❹ Step 3: Read the data from Port D



# Examples (Cont..)

- ❶ Example 3: Make upper nibble of PORTD as output and lower nibble as input. Out 3 on upper nibble and read back data using PIND
- ❷ Step 1: Make Port Ds Pin7 to Pin4 as output pins and Pin3 to Pin0 as input pins

DDRD =

D7	D6	D5	D4	D3	D2	D1	D0
1	1	1	1	0	0	0	0

DDRD = 0xF0

- ❸ Step 2: Put data on Port Ds Pin7 to Pin4 and enable pullup for Pin3 to Pin0

PORTD =

D7	D6	D5	D4	D3	D2	D1	D0
0	0	1	1	1	1	1	1

PORTD = 0x3F

- ❹ Step 3: Read the data from Port D

x = PIND;



# Examples (Cont..)

- ❶ Example 3: Make upper nibble of PORTD as output and lower nibble as input. Out 3 on upper nibble and read back data using PIND
- ❷ Step 1: Make Port Ds Pin7 to Pin4 as output pins and Pin3 to Pin0 as input pins

$$\text{DDRD} =$$

D7	D6	D5	D4	D3	D2	D1	D0
1	1	1	1	0	0	0	0

$\text{DDRD} = 0xF0$

- ❸ Step 2: Put data on Port Ds Pin7 to Pin4 and enable pullup for Pin3 to Pin0

$$\text{PORTD} =$$

D7	D6	D5	D4	D3	D2	D1	D0
0	0	1	1	1	1	1	1

$\text{PORTD} = 0x3F$

- ❹ Step 3: Read the data from Port D

$x = \text{PIND};$

Therefore value of x will be 0x3F



# Buzzer Interfacing in Firebird V





# Buzzer Interfacing in Firebird V

## ① Buzzer Connected to PortC pin 3



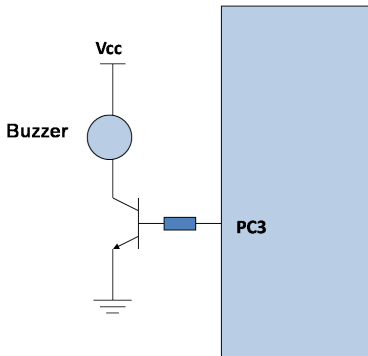
# Buzzer Interfacing in Firebird V

## 1 Buzzer Connected to PortC pin 3



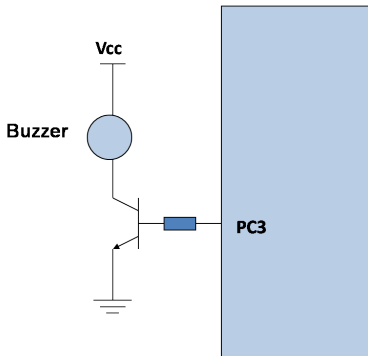
# Buzzer Interfacing in Firebird V

## 1 Buzzer Connected to PortC pin 3



# Buzzer Interfacing in Firebird V

## 1 Buzzer Connected to PortC pin 3

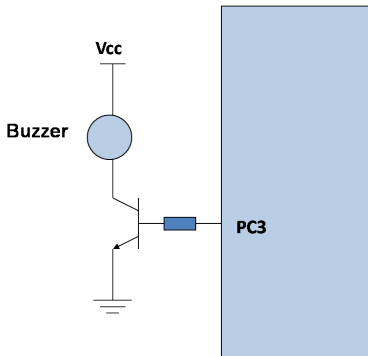


## 2 To Turn on buzzer:



# Buzzer Interfacing in Firebird V

## 1 Buzzer Connected to PortC pin 3

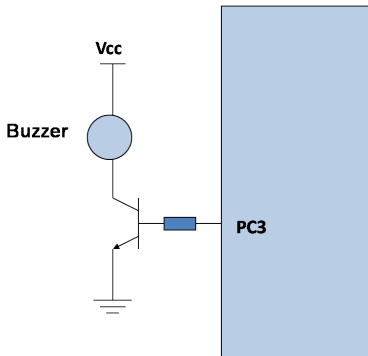


## 2 To Turn on buzzer:



# Buzzer Interfacing in Firebird V

## 1 Buzzer Connected to PortC pin 3

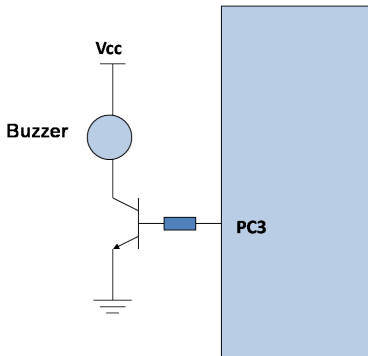


## 2 To Turn on buzzer: send logic HIGH on pin3 of PortC



# Buzzer Interfacing in Firebird V

## 1 Buzzer Connected to PortC pin 3



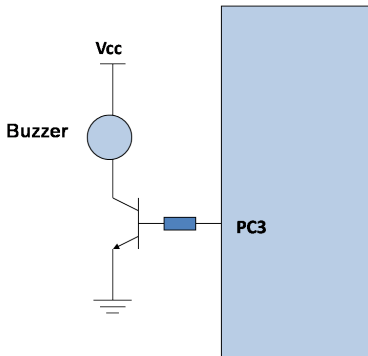
2 To Turn on buzzer: send logic HIGH on pin3 of PortC

3 To Turn off buzzer:



# Buzzer Interfacing in Firebird V

## 1 Buzzer Connected to PortC pin 3



2 To Turn on buzzer: send logic HIGH on pin3 of PortC

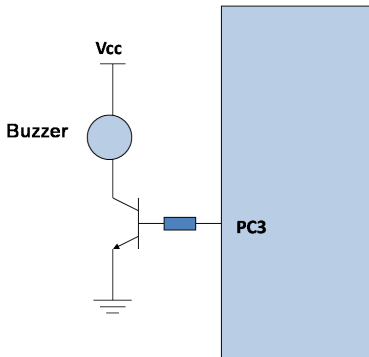
3 To Turn off buzzer:





# Buzzer Interfacing in Firebird V

## 1 Buzzer Connected to PortC pin 3



2 To Turn on buzzer: send logic HIGH on pin3 of PortC

3 To Turn off buzzer: send logic LOW on pin3 of PortC



# Buzzer Program



# Buzzer Program

- 1 Configure PORTC3,pin as output.



# Buzzer Program

- 1 Configure PORTC3,pin as output.



# Buzzer Program

- 1 Configure PORTC3,pin as output.

DDRC=



# Buzzer Program

- 1 Configure PORTC3,pin as output.

```
DDRC= 0x08; //0000 1000
```



# Buzzer Program

- 1 Configure PORTC3,pin as output.

```
DDRC= 0x08; //0000 1000
```

- 2 To turn ON the buzzer set PC3 output high



# Buzzer Program

- 1 Configure PORTC3,pin as output.

```
DDRC= 0x08; //0000 1000
```

- 2 To turn ON the buzzer set PC3 output high





# Buzzer Program

- 1 Configure PORTC3,pin as output.

```
DDRC= 0x08; //0000 1000
```

- 2 To turn ON the buzzer set PC3 output high

```
PORTC =
```



# Buzzer Program

- 1 Configure PORTC3,pin as output.

```
DDRC= 0x08; //0000 1000
```

- 2 To turn ON the buzzer set PC3 output high

```
PORTC = 0x08; //0000 1000
```



# Buzzer Program

- 1 Configure PORTC3,pin as output.

```
DDRC= 0x08; //0000 1000
```

- 2 To turn ON the buzzer set PC3 output high

```
PORTC = 0x08; //0000 1000
```

- 3 To turn OFF the buzzer set PC3 output low



# Buzzer Program

- 1 Configure PORTC3,pin as output.

```
DDRC= 0x08; //0000 1000
```

- 2 To turn ON the buzzer set PC3 output high

```
PORTC = 0x08; //0000 1000
```

- 3 To turn OFF the buzzer set PC3 output low



# Buzzer Program

- 1 Configure PORTC3,pin as output.

```
DDRC= 0x08; //0000 1000
```

- 2 To turn ON the buzzer set PC3 output high

```
PORTC = 0x08; //0000 1000
```

- 3 To turn OFF the buzzer set PC3 output low

```
PORTC =
```



# Buzzer Program

- 1 Configure PORTC3,pin as output.

```
DDRC= 0x08; //0000 1000
```

- 2 To turn ON the buzzer set PC3 output high

```
PORTC = 0x08; //0000 1000
```

- 3 To turn OFF the buzzer set PC3 output low

```
PORTC = 0x00; //0000 0000
```



# Buzzer Program

- 1 Configure PORTC3,pin as output.

```
DDRC= 0x08; //0000 1000
```

- 2 To turn ON the buzzer set PC3 output high

```
PORTC = 0x08; //0000 1000
```

- 3 To turn OFF the buzzer set PC3 output low

```
PORTC = 0x00; //0000 0000
```



# AVR Programming Tools





# AVR Programming Tools

## ① Software Required.



# AVR Programming Tools

## ① Software Required.



# AVR Programming Tools

## ① Software Required.

ATMEL STUDIO 6



# AVR Programming Tools

## ① Software Required.

### ATMEL STUDIO 6

- Integrated Development Environment (IDE)
- Supports Developing and Debugging of AVR and ARM based microcontroller application
- Download Link:  
[www.atmel.com/Microsite/atmel\\_studio6/default.aspx](http://www.atmel.com/Microsite/atmel_studio6/default.aspx)



# AVR Programming Tools

## ① Software Required.

### ATMEL STUDIO 6

- Integrated Development Environment (IDE)
- Supports Developing and Debugging of AVR and ARM based microcontroller application
- Download Link:  
[www.atmel.com/Microsite/atmel\\_studio6/default.aspx](http://www.atmel.com/Microsite/atmel_studio6/default.aspx)

## ② Hardware Required



# AVR Programming Tools

## ❶ Software Required.

### ATMEL STUDIO 6

- Integrated Development Environment (IDE)
- Supports Developing and Debugging of AVR and ARM based microcontroller application
- Download Link:  
[www.atmel.com/Microsite/atmel\\_studio6/default.aspx](http://www.atmel.com/Microsite/atmel_studio6/default.aspx)

## ❷ Hardware Required



# AVR Programming Tools

## ① Software Required.

### ATMEL STUDIO 6

- Integrated Development Environment (IDE)
- Supports Developing and Debugging of AVR and ARM based microcontroller application
- Download Link:  
[www.atmel.com/Microsite/atmel\\_studio6/default.aspx](http://www.atmel.com/Microsite/atmel_studio6/default.aspx)

## ② Hardware Required

- hex file can be loaded into microcontroller using



# AVR Programming Tools

## ① Software Required.

### ATMEL STUDIO 6

- Integrated Development Environment (IDE)
- Supports Developing and Debugging of AVR and ARM based microcontroller application
- Download Link:  
[www.atmel.com/Microsite/atmel\\_studio6/default.aspx](http://www.atmel.com/Microsite/atmel_studio6/default.aspx)

## ② Hardware Required

- hex file can be loaded into microcontroller using  
a. Bootloader





# AVR Programming Tools

## ① Software Required.

### ATMEL STUDIO 6

- Integrated Development Environment (IDE)
- Supports Developing and Debugging of AVR and ARM based microcontroller application
- Download Link:  
[www.atmel.com/Microsite/atmel\\_studio6/default.aspx](http://www.atmel.com/Microsite/atmel_studio6/default.aspx)

## ② Hardware Required

- hex file can be loaded into microcontroller using
  - a. Bootloader



# AVR Programming Tools

## ① Software Required.

### ATMEL STUDIO 6

- Integrated Development Environment (IDE)
- Supports Developing and Debugging of AVR and ARM based microcontroller application
- Download Link:  
[www.atmel.com/Microsite/atmel\\_studio6/default.aspx](http://www.atmel.com/Microsite/atmel_studio6/default.aspx)

## ② Hardware Required

- hex file can be loaded into microcontroller using
  - a. Bootloader
  - b. AVR Programmers viz. AVR MKII, AVRdude, Pony-Programmer, etc.



# Syntax for C-Program



# Syntax for C-Program

```
#include
```



# Syntax for C-Program

#include

```
#include <avr/io.h>
```

```
#include <avr/interrupt.h>
```

```
#include <util/delay.h>
```



# Syntax for C-Program

#include

```
#include <avr/io.h>
```

```
#include <avr/interrupt.h>
```

```
#include <util/delay.h>
```

Pin Configuration



# Syntax for C-Program

## #include

```
#include <avr/io.h>
#include <avr/interrupt.h>
#include <util/delay.h>
```

## Pin Configuration

```
void buzzer_pin_config (void)
{
    DDRC =
    PORTC =
}
```



# Syntax for C-Program





# Syntax for C-Program

Main-Program



# Syntax for C-Program

## Main-Program

```
int main (void)
{
    buzzer_pin_config ();
    while(1)
    {
        buzzer_on();
        _delay_ms();
        buzzer_off();
        _delay_ms();
    }
}
```



# Syntax for C-Program

## Main-Program

```
int main (void)
{
    buzzer_pin_config ();
    while(1)
    {
        buzzer_on();
        _delay_ms();
        buzzer_off();
        _delay_ms();
    }
}
```

## Functions



# Syntax for C-Program

## Main-Program

```
int main (void)
{
    buzzer_pin_config ();
    while(1)
    {
        buzzer_on();
        _delay_ms();
        buzzer_off();
        _delay_ms();
    }
}
```

## Functions

```
void buzzer_on (void)
{
    PORTC =
```

---



# Syntax for C-Program

## Main-Program

```
int main (void)
{
    buzzer_pin_config ();
    while(1)
    {
        buzzer_on();
        _delay_ms();
        buzzer_off();
        _delay_ms();
    }
}
```

## Functions

```
void buzzer_on (void)
{
    PORTC =
```

---

```
void buzzer_off (void)
{
    PORTC =
```



# Thank You!

Post your queries on: <http://qa.e-yantra.org/>

