

## 5: BASIC Stamp Command Reference – OUTPUT

### OUTPUT

BS1	BS2	BS2e	BS2sx	BS2p	BS2pe	BS2px
-----	-----	------	-------	------	-------	-------



#### OUTPUT *Pin*

#### Function

Make the specified pin an output.

- ***Pin*** is a variable/constant/expression (0 – 15) that specifies which I/O pin to set to output mode.

#### Quick Facts

**Table 5.62:** OUTPUT Quick Facts.

	BS1 and all BS2 Models
Related Commands	INPUT and REVERSE

#### Explanation

There are several ways to make a pin an output. Commands that rely on output pins, like PULSOUT and SEROUT, automatically change the specified pin to output. Writing 1s to particular bits of the variable DIRS makes the corresponding pins outputs. And then there's the OUTPUT command.

When a pin is an output, your program can change its state by writing to the corresponding bit in the OUTS variable (PINS on the BS1). For example:



```
OUTPUT 4
OUT4 = 1
```

EFFECTS OF SETTING AN INPUT PIN TO AN OUTPUT.

When your program changes a pin from input to output, whatever state happens to be in the corresponding bit of OUTS (PINS on the BS1) sets the initial state of the pin. To simultaneously make a pin an output and set its state use the HIGH and LOW commands.



#### Demo Program (INPUT\_OUTPUT.bs1)

```
' INPUT_OUTPUT.bs1
' This program demonstrates how the input/output direction of a pin is
' determined by the corresponding bit of DIRS. It also shows that the
' state of the pin itself (as reflected by the corresponding bit of PINS)
' is determined by the outside world when the pin is an input, and by the
' corresponding bit of PINS when it's an output. To set up the demo,
' connect a 10k resistor from +5V to P7 on the BASIC Stamp. The resistor
```

## OUTPUT – BASIC Stamp Command Reference

---

```
' to +5V puts a high (1) on the pin when it's an input. The BASIC Stamp
' can override this state by writing a low (0) to bit 7 of OUTS and
' changing the pin to output.

' {$STAMP BS1}
' {$PBASIC 1.0}

Main:
  INPUT 7                                ' Make P7 an input
  DEBUG "State of P7: ", #PIN7, CR

  PIN7 = 0                               ' Write 0 to output latch
  DEBUG "After 0 written to OUT7: "
  DEBUG #PIN7, CR

  OUTPUT 7                               ' Make P7 an output
  DEBUG "After P7 changed to output: "
  DEBUG #PIN7
END
```

### Demo Program (INPUT\_OUTPUT.bs2)

```
' INPUT_OUTPUT.bs2
' This program demonstrates how the input/output direction of a pin is
' determined by the corresponding bit of DIRS. It also shows that the
' state of the pin itself (as reflected by the corresponding bit of INS)
' is determined by the outside world when the pin is an input, and by the
' corresponding bit of OUTS when it's an output. To set up the demo,
' connect a 10k resistor from +5V to P7 on the BASIC Stamp. The resistor
' to +5V puts a high (1) on the pin when it's an input. The BASIC Stamp
' can override this state by writing a low (0) to bit 7 of OUTS and
' changing the pin to output.

' {$STAMP BS2}
' {$PBASIC 2.5}

Main:
  INPUT 7                                ' Make P7 an input
  DEBUG "State of P7: ",
    BIN1 IN7, CR

  OUT7 = 0                               ' Write 0 to output latch
  DEBUG "After 0 written to OUT7: ",
    BIN1 IN7, CR

  OUTPUT 7                               ' Make P7 an output
  DEBUG "After P7 changed to output: ",
    BIN1 IN7
END
```



NOTE: This example program can be used with all BS2 models by changing the \$STAMP directive accordingly.