# Fixed loops



**LEARNING INTENTION**

We are learning to use fixed loops to repeat steps a set number of times.

**SUCCESS CRITERIA — I CAN**

- ☐ name a fixed loop and a continuous loop
- ☐ draw a fixed loop with a counting decision
- ☐ loop back to the correct step
- ☐ read a program with a counter

## Fixed and continuous loops

| Loop | Repeats | Ends? |
|---|---|---|
| Fixed | a set number of times (e.g. 5) | Yes, then the program carries on |
| Continuous | forever | No |

**SHORT CIRCUIT**

**× WHAT EXAMINERS SAW**

Asked to name the type of loop, most candidates did not state 'fixed loop'. Common wrong answers were 'open loop', 'closed loop' or 'continuous loop'.

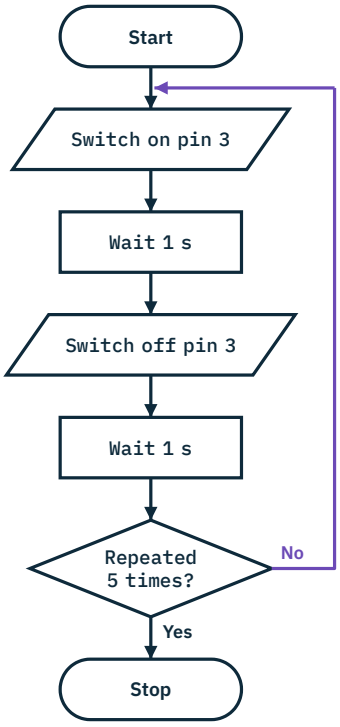
**√ THE FIX**

A loop that repeats a set number of times is a fixed loop. Open and closed loop are control systems, not program loops.

Course report 2025, Question 2(c).

# Fixed loops

## Drawing a fixed loop




Flash 5 times

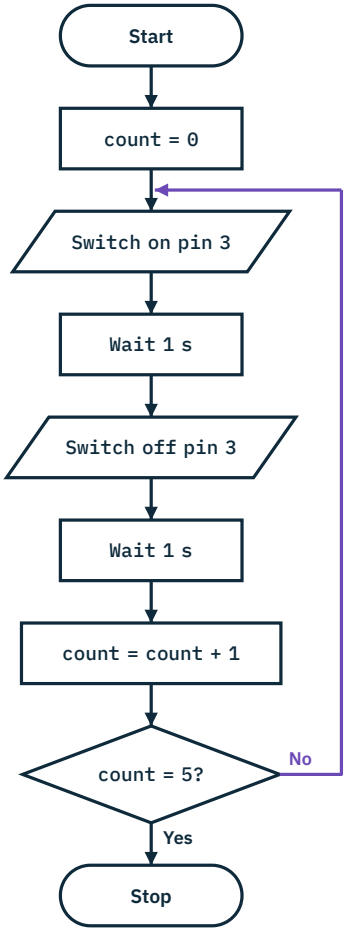
Put a decision after the steps that repeat:
**Repeated 5 times?**

The No route loops back to the first repeated step. Yes carries on to the next step (here, Stop).

In the exam you may write '×5?' or 'Repeated 5 times?'.

# Fixed loops

## Inside the microcontroller: a counter




The same loop with a counter

A counter is a number stored in the microcontroller. It starts at 0 and goes up by 1 each time round the loop.

When it reaches the target, the decision answers Yes.

# Fixed loops





WORKED EXAMPLE **How many flashes?**

| 1 GIVEN | 2 FIND | 3 FORMULA | 4 SUBSTITUTE | 5 ANSWER + UNIT |
|---|---|---|---|---|
| let count = 0<br>loop:<br>switch on,<br>pause,<br>switch off,<br>pause<br>let count = count + 1<br>if count = 15 then end | number of flashes | count goes up by 1 each time round<br>stops when count = 15 | count after loops: 1, 2, … 15 | **15 flashes** |

The loop is a fixed loop: it repeats a set number of times, then the program moves on.

In code this is often called a **for...next** loop. The three languages below all flash pin 3 five times.

**PICAXE BASIC**

```
for b0 = 1 to 5
  high B.3
  pause 1000
  low B.3
  pause 1000
next b0
```

**Arduino C++**

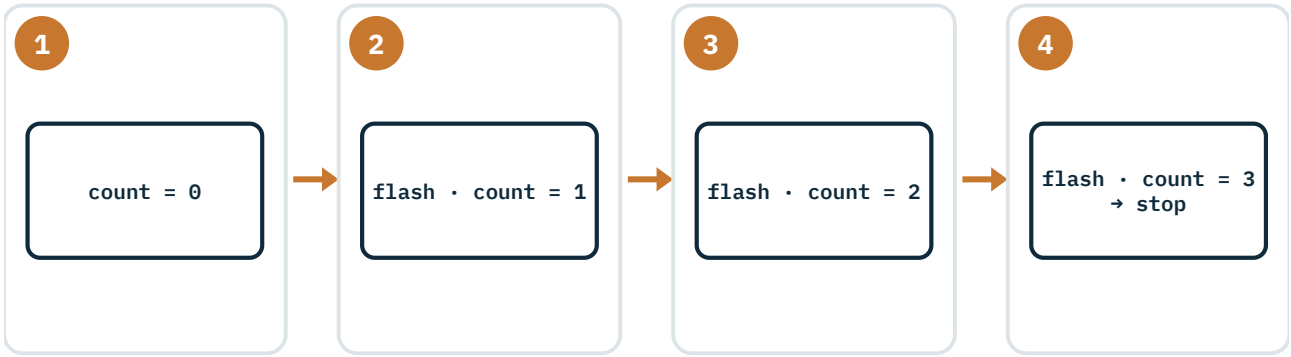
```
for (int i = 0; i < 5; i++) {
  digitalWrite(3, HIGH);
  delay(1000);
  digitalWrite(3, LOW);
  delay(1000);
}
```

**micro:bit Python**

```
for i in range(5):
    pin3.write_digital(1)
    sleep(1000)
    pin3.write_digital(0)
    sleep(1000)
```

## Tracing a loop

TRACING A FIXED LOOP




Before the loop | Pass 1 | Pass 2 | Pass 3: count = 3, carry on

# Fixed loops



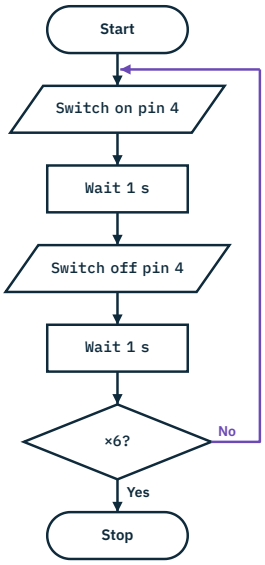
WORKED EXAMPLE **Trace table for a counter**

| 1 GIVEN | 2 FIND | 3 FORMULA | 4 SUBSTITUTE | 5 ANSWER + UNIT |
|---|---|---|---|---|
| count = 0; loop: flash; count = count + 1; if count = 3 then end | count values | add 1 each pass | pass 1: count = 1<br>pass 2: count = 2<br>pass 3: count = 3 → end | **3 flashes** |

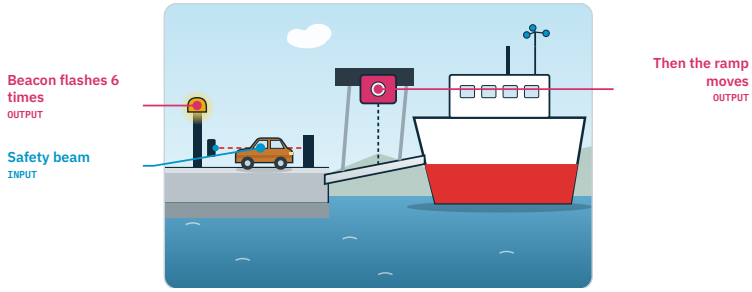
WORKED EXAMPLE **Where does the loop go back to?**

| 1 GIVEN | 2 FIND | 3 FORMULA | 4 SUBSTITUTE | 5 ANSWER + UNIT |
|---|---|---|---|---|
| steps: pin 4 on, wait 1 s, pin 4 off, wait 1 s, ×6? | loop target | back to the first repeated step | the first repeated step is 'Switch on pin 4' | **Arrow to just above 'Switch on pin 4'** |

Not back to Start.




Fixed loop: ×6



# Fixed loops

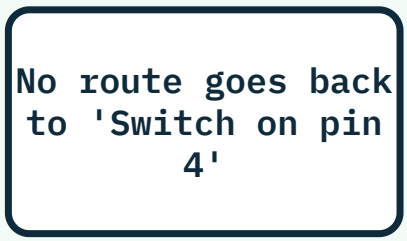


**WRONG**




Back to Start repeats things that should run once.

**RIGHT**




Loop back to the first repeated step.

## REAL ENGINEERING · STRATHMORE FERRY RAMP HORN

Before the Strathmore (fictional) ramp moves, the horn sounds three short blasts.

A fixed loop of 3 makes the warning the same every time, then the program moves on to lower the ramp.

## PROBE · QUICK CHECKS

1. Loop that repeats a set number of times? _____
2. Where does the No route of '×5?' go? _____
3. Is 'closed loop' a type of program loop? _____

## EXAM READY · 8 MARKS

1 **State** the name of the type of loop used to repeat steps 20 times. **[1]**

2 A program contains: let count = 0 … let count = count + 1 … if count = 8 then label_3. **[1]**
**State** how many times the loop runs.

# Fixed loops

3 A beacon on pin 2 flashes 10 times (on 0.5 s, off 0.5 s), then a lamp on pin 4 stays on. **[6]**

**Complete** the flowchart.

