

```

-----MODULE_FGBoundedBuffer-----
EXTENDS Integers, Sequences, ISequences

CONSTANT N, Msg, Input
ASSUME /\ N \in Nat \ {0}
        /\ Input \in ISeq(Msg)

a(+)b == (a+b)%2*N
a(-)b == (a-b)%2*N

--algorithm_FGBBuf{
  variables in = Input, out = <<>,
  buf \in [0..(N-1) -> Msg], p = 0, c = 0;

  process (Producer = "P")
  {
    p1: while (TRUE)
    {
      await p(-) < c # N;
      p2: buf[p % N] := IHead(in);
      in := ITail(in);
      p3: p := p(+) 1
    }
  }

  fair process (Consumer = "C")
  {
    c1: while (TRUE)
    {
      await p # c;
      c2: out := Append(out, buf[c % N]);
      c3: c := c(+) 1
    }
  }
}

*****
=====

```