```
proc acelerar (inout r: reunion, in prof: ℤ, in freq: ℤ) {
    Pre {
      r_0 = r ∧
        esReuniónVálidaAux(r, prof, freq) ∧
        lasSeñalesDuranMásDe2Segundos(r)
    }
    Post {
      |r| = |r_0| ∧_L
        lasSeñalesTieneLaMitadDeMuestras(r, r_0) ∧_L
        losImpares(r, r_0)
    }
}
  pred lasSeñalesDuranMásDe2Segundos (r: reunion, freq: ℤ) {
    (|r| > 0 ∧ freq ≠ 0) ∧_L duraciónEnSegundos(r[0]_0, freq) > 2
  }

  pred lasSeñalesTieneLaMitadDeMuestras (r: reunion, r_v: reunion) {
    (∀i : ℤ) 0 ≤ i < |r| ⟶_L if esPar(|r_v[i]_0|) then  else  fi|r[i]_0| = |r_v[i]_0| / 2  |r[i]_0| = |r_v[i]_0| − 1 / 2
  }

  pred losImpares (r: reunion, r_v: reunion) {
    (∀i : ℤ) 0 ≤ i < |r_v| ⟶_L (
      (∃j : ℤ) 0 ≤ j < |r| ∧_L ( r_v[i]_1 = r[j]_1 ) ∧_L (
        (∀q : ℤ) 0 ≤ q < |r_v[i]_0| ∧ (¬esPar(q)) ⟶_L ( r_v[i]_0[q] = r[j]_0[(q−1)/2] ) ) )
  }
```