ecl-es-halt[11,40]

```
ecl-es-halt(es; x)
≡def ecl_ind(x;
            k,test.(λn,e1,e2. if (n =0 0)
            then e2 =
                  first e ≥ e1.(es-kind(es; e) = k)
                              c∧ (↑(test(es-state-when(es; e),es-val(es; e))))
            else False
            fi );
            a,b,ha,hb.(λn,e1,e2. if (n =0 0) then False else ha(n,e1,e2) fi
            ∨ ∃e∈(e1,e2].(ha(0,e1,es-pred(es; e))) ∧ (hb(n,e,e2)));
            a,b,ha,hb.(λn,e1,e2. if (n =0 0)
            then ((ha(0,e1,e2)) ∧ ∃e∈[e1,e2].hb(0,e1,e))
                  ∨ ((hb(0,e1,e2)) ∧ ∃e∈[e1,e2].ha(0,e1,e))
            else ((ha(n,e1,e2))
                 ∧ l-all(cons(0; ecl-ex(b));
                         m.if n ≤z m
                         then ∀e∈[e1,e2).¬(hb(m,e1,e))
                         else ∀e∈[e1,e2].¬(hb(m,e1,e))
                         fi ))
                 ∨ ((hb(n,e1,e2))
                   ∧ l-all(cons(0; ecl-ex(a));
                           m.if n ≤z m
                           then ∀e∈[e1,e2).¬(ha(m,e1,e))
                           else ∀e∈[e1,e2].¬(ha(m,e1,e))
                           fi ))
            fi );
            a,b,ha,hb.(λn,e1,e2. ((ha(n,e1,e2))
            ∧ l-all(cons(0; ecl-ex(b));
                    m.if n ≤z m
                    then ∀e∈[e1,e2).¬(hb(m,e1,e))
                    else ∀e∈[e1,e2].¬(hb(m,e1,e))
                    fi ))
            ∨ ((hb(n,e1,e2))
              ∧ l-all(cons(0; ecl-ex(a));
                      m.if n ≤z m
                      then ∀e∈[e1,e2).¬(ha(m,e1,e))
                      else ∀e∈[e1,e2].¬(ha(m,e1,e))
                      fi )));
            a,ha.(λn,e1,e2. if (n =0 0)
            then False
            else [e1;e2]∼([x,y].ha(0,x,y))∗[x,y].ha(n,x,y)
            fi );
```

$a,m,ha.ha;$
$a,m,ha.(\lambda n,e_1,e_2.$ if $(n =_0 0)$ then False else $ha(n,e_1,e_2)$ fi
$\vee$ if $(n =_0 m)$ then $ha(0,e_1,e_2)$ else False fi);
$a,l,ha.(\lambda n,e_1,e_2.\ ((ha(n,e_1,e_2)) \wedge (\neg(n \in l)))$
$\vee$ if $(n =_0 0)$ then l_exists(l; $\mathbb{N}$; $m.(ha(m,e_1,e_2))$) else False fi))

clarification:

ecl-es-halt(es; x)
$\equiv_{\text{def}}$ ecl_ind(x;
$k,test.(\lambda n,e_1,e_2.$ if $(n =_0 0)$
then es-first-since(es;e.(es-kind(es; e) $= k \in$ Knd)
c$\wedge$ ($\uparrow$($test$(es-state-when(es; e),es-val(es; e))));e_1;e_2)
else False
fi);
$a,b,ha,hb.(\lambda n,e_1,e_2.$ if $(n =_0 0)$ then False else $ha(n,e_1,e_2)$ fi
$\vee$ existse-between3(es;e_1;e_2;e.(ha(0,e_1,es-pred(es; e))) $\wedge$ ($hb(n,e,e_2)$)));
$a,b,ha,hb.(\lambda n,e_1,e_2.$ if $(n =_0 0)$
then ((ha(0,e_1,e_2)) $\wedge$ existse-between2(es;e_1;e_2;$e.hb$(0,e_1,e)))
$\vee$ ((hb(0,e_1,e_2)) $\wedge$ existse-between2(es;e_1;e_2;$e.ha$(0,e_1,e)))
else (($ha(n,e_1,e_2)$)
$\wedge$ l-all(cons(0; ecl-ex(b));
m.if $n \leq$z m
then alle from es in [e_1;e_2).$\neg(hb(m,e_1,e)$)
else alle-between2(es;e_1;e_2;$e.\neg(hb(m,e_1,e)$))
fi))
$\vee$ (($hb(n,e_1,e_2)$)
$\wedge$ l-all(cons(0; ecl-ex(a));
m.if $n \leq$z m
then alle from es in [e_1;e_2).$\neg(ha(m,e_1,e)$)
else alle-between2(es;e_1;e_2;$e.\neg(ha(m,e_1,e)$))
fi))
fi);
$a,b,ha,hb.(\lambda n,e_1,e_2.$ (($ha(n,e_1,e_2)$)
$\wedge$ l-all(cons(0; ecl-ex(b));
m.if $n \leq$z m
then alle from es in [e_1;e_2).$\neg(hb(m,e_1,e)$)
else alle-between2(es;e_1;e_2;$e.\neg(hb(m,e_1,e)$))
fi))
$\vee$ (($hb(n,e_1,e_2)$)
$\wedge$ l-all(cons(0; ecl-ex(a));
m.if $n \leq$z m
then alle from es in [e_1;e_2).$\neg(ha(m,e_1,e)$)
else alle-between2(es;e_1;e_2;$e.\neg(ha(m,e_1,e)$))
fi)));
$a,ha.(\lambda n,e_1,e_2.$ if $(n =_0 0)$

then False
else es-pstar-q(es;x,y.ha(0,x,y);x,y.ha(n,x,y);e_1;e_2)
fi);
a,m,ha.ha;
a,m,ha.(λn,e_1,e_2. if ($n =_0 0$) then False else ha(n,e_1,e_2) fi
$\vee$ if ($n =_0 m$) then ha(0,e_1,e_2) else False fi);
a,l,ha.(λn,e_1,e_2. ((ha(n,e_1,e_2)) $\wedge$ ($\neg$($n \in l \in \mathbb{N}$)))
$\vee$ if ($n =_0 0$) then l_exists(l; $\mathbb{N}$; m.(ha(m,e_1,e_2))) else False fi))

http://www.nuprl.org/FDLcontent/p0_963683_/p44_265469_{ecl-es-halt}.html