conditional-send-p[11,40]

k(v:B) sends on l [tg:T, f <state, v>]?[]
$\equiv_{\text{def}}$ ((∀x:Id. vartype(source(l);x) ⊆r ds(x)?Top)
& (∀e:E. (loc(e) = source(l)) ⇒ (kind(e) = k) ⇒ (valtype(e) ⊆r B))
& (∀e:E. (kind(e) = rcv(l,tg)) ⇒ (valtype(e) ⊆r T)))
c∧ (∀e:E.
(loc(e) = source(l))
⇒ (kind(e) = k)
⇒ (((↑can-apply(f;<(state when e), val(e)>))
⇒ (∃e':E
((kind(e') = rcv(l,tg))
c∧ (sender(e') = e
& (∀e'':E.
(kind(e'') = rcv(l,tg)) ⇒ (sender(e'') = e) ⇒ (e'' = e'))
& val(e') = do-apply(f;<(state when e), val(e)>)))))
& ((¬(↑can-apply(f;<(state when e), val(e)>)))
⇒ (¬(∃e':E. ((kind(e') = rcv(l,tg)) c∧ (sender(e') = e)))))))

clarification:

conditional-send-p(es;ds;k;B;l;tg;T;f)
$\equiv_{\text{def}}$ ((∀x:Id. es-vartype(es; source(l); x) ⊆r fpf-cap(ds;IdDeq;x;Top))
& (∀e:es-E(es).
(es-loc(es; e) = source(l) ∈ Id)
⇒ (es-kind(es; e) = k ∈ Knd)
⇒ (es-valtype(es; e) ⊆r B))
& (∀e:es-E(es). (es-kind(es; e) = rcv(l,tg) ∈ Knd) ⇒ (es-valtype(es; e) ⊆r T)))
c∧ (∀e:es-E(es).
(es-loc(es; e) = source(l) ∈ Id)
⇒ (es-kind(es; e) = k ∈ Knd)
⇒ (((↑can-apply(f;<es-state-when(es;e), es-val(es; e)>))
⇒ (∃e':es-E(es)
((es-kind(es; e') = rcv(l,tg) ∈ Knd)
c∧ (es-sender(es; e') = e ∈ es-E(es)
& (∀e'':es-E(es).
(es-kind(es; e'') = rcv(l,tg) ∈ Knd)
⇒ (es-sender(es; e'') = e ∈ es-E(es))
⇒ (e'' = e' ∈ es-E(es)))
& es-val(es; e')
=
do-apply(f;<es-state-when(es;e), es-val(es; e)>)
∈ T))))
& ((¬(↑can-apply(f;<es-state-when(es;e), es-val(es; e)>)))

$\Rightarrow$ ($\neg$($\exists e'$:es-E(es)
((es-kind(es; e') = rcv(l,tg) $\in$ Knd)
c$\wedge$ (es-sender(es; e') = $e \in$ es-E(es)))))))

http://www.nuprl.org/FDLcontent/p0_963683_/p27_308273_{conditional-send-p}.html