```
recognizer1__realizes  THM

⊢  ∀T,A:Type, P:(A→T→𝔹), k:Knd, i,r,x:Id.
|  A
|  ⇒
|  T
|  ⇒
|  ¬x = r
|  ⇒
|  λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))
|  realizes es.∀e@i.kind(e) = k ⇒ (valtype(e) ⊆r T)
|              & (vartype(i;x) ⊆r A)
|              & ∀e@i.first(e) ⇒ (r when e) = false2
|              & ∀e'@i.(r after e')
|                       ⟺
|                       (∃e:E.
|                        (e <loc e') ∨ e = e'
|                        & kind(e) = k
|                        & P((x when e),val(e)))
| by RealizerSetup
|    THEN
|    Assert
|    ((vartype(i;r) ⊆r 𝔹) & ∀e@i.first(e) ⇒ (r when e) = false2)
|
|\
| .....assertion .....
| 1. T : Type
| 2. A : Type
| 3. P : A→T→𝔹
| 4. k : Knd
| 5. i : Id
| 6. r : Id
| 7. x : Id
| 8. A
| 9. T
| 10. ¬x = r
| 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
| 12. D' : Dsys
| 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
| 14. w : World
| 15. p : FairFifo
| 16. PossibleWorld(D';w)
| ⊢  (vartype(i;r) ⊆r 𝔹) & ∀e@i.first(e) ⇒ (r when e) = false2
|   by InitRule
|
.\
. 1. T : Type
. 2. A : Type
. 3. P : A→T→𝔹
. 4. k : Knd
. 5. i : Id
. 6. r : Id
. 7. x : Id
. 8. A
. 9. T
```

```
. 10. ¬x = r
. 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. 12. D' : Dsys
. 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. 14. w : World
. 15. p : FairFifo
. 16. PossibleWorld(D';w)
. 17. (vartype(i;r) ⊆r 𝔹) & ∀e@i.first(e) ⇒ (r when e) = false2
. ⊢  ∀e@i.kind(e) = k ⇒ (valtype(e) ⊆r T) & (vartype(i;x) ⊆r A)
. |  & ∀e@i.first(e) ⇒ (r when e) = false2
. |  & ∀e'@i.(r after e')
. |          ⟺
. |          (∃e:E.
. |           (e <loc e') ∨ e = e'
. |           & kind(e) = k
. |           & P((x when e),val(e)))
. | by Assert ∀e@i.¬(r after e) = (r when e) ⇒ kind(e) = k
. |
. |\
. | .....assertion .....
. | 1. T : Type
. | 2. A : Type
. | 3. P : A→T→𝔹
. | 4. k : Knd
. | 5. i : Id
. | 6. r : Id
. | 7. x : Id
. | 8. A
. | 9. T
. | 10. ¬x = r
. | 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. | 12. D' : Dsys
. | 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. | 14. w : World
. | 15. p : FairFifo
. | 16. PossibleWorld(D';w)
. | 17. (vartype(i;r) ⊆r 𝔹) & ∀e@i.first(e) ⇒ (r when e) = false2
. | ⊢  ∀e@i.¬(r after e) = (r when e) ⇒ kind(e) = k  by FrameRule
. |
. .\
. . 1. T : Type
. . 2. A : Type
. . 3. P : A→T→𝔹
. . 4. k : Knd
. . 5. i : Id
. . 6. r : Id
. . 7. x : Id
. . 8. A
. . 9. T
. . 10. ¬x = r
. . 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . 12. D' : Dsys
. . 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . 14. w : World
. . 15. p : FairFifo
```

```
. . 16. PossibleWorld(D';w)
. . 17. (vartype(i;r) ⊆r 𝔹) & ∀e@i.first(e) ⇒ (r when e) = false2
. . 18. ∀e@i.¬(r after e) = (r when e) ⇒ kind(e) = k
. . ⊢  ∀e@i.kind(e) = k ⇒ (valtype(e) ⊆r T) & (vartype(i;x) ⊆r A)
. . |  & ∀e@i.first(e) ⇒ (r when e) = false2
. . |  & ∀e'@i.(r after e')
. . |          ⟺
. . |          (∃e:E.
. . |           (e <loc e') ∨ e = e'
. . |           & kind(e) = k
. . |           & P((x when e),val(e)))
. . | by EffectRule1 i r x k T 𝔹 A (λr,x,v. P(x,v) ∨2 r)
. . |
. . 1. T : Type
. . 2. A : Type
. . 3. P : A→T→𝔹
. . 4. k : Knd
. . 5. i : Id
. . 6. r : Id
. . 7. x : Id
. . 8. A
. . 9. T
. . 10. ¬x = r
. . 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . 12. D' : Dsys
. . 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . 14. w : World
. . 15. p : FairFifo
. . 16. PossibleWorld(D';w)
. . 17. vartype(i;r) ⊆r 𝔹
. . 18. ∀e@i.first(e) ⇒ (r when e) = false2
. . 19. ∀e@i.¬(r after e) = (r when e) ⇒ kind(e) = k
. . 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v) ∨2 r) ∈ Dsys
. . 21. vartype(i;r) ⊆r 𝔹
. . 22. vartype(i;x) ⊆r A
. . 23. ∀e:E. loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . 24. ∀e:E.
. . |   loc(e) = i
. . |   ⇒
. . |   kind(e) = k
. . |   ⇒
. . |   (r after e) = (P((x when e),val(e)) ∨2 (r when e))
. . ⊢  ∀e@i.kind(e) = k ⇒ (valtype(e) ⊆r T) & (vartype(i;x) ⊆r A)
. . |  & ∀e@i.first(e) ⇒ (r when e) = false2
. . |  & ∀e'@i.(r after e')
. . |          ⟺
. . |          (∃e:E.
. . |           (e <loc e') ∨ e = e'
. . |           & kind(e) = k
. . |           & P((x when e),val(e)))  by Analyze 0
. . |
. . |\
. . | 1. T : Type
. . | 2. A : Type
. . | 3. P : A→T→𝔹
```

```
. . | 4. k : Knd
. . | 5. i : Id
. . | 6. r : Id
. . | 7. x : Id
. . | 8. A
. . | 9. T
. . | 10. ¬x = r
. . | 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . | 12. D' : Dsys
. . | 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . | 14. w : World
. . | 15. p : FairFifo
. . | 16. PossibleWorld(D';w)
. . | 17. vartype(i;r) ⊆r 𝔹
. . | 18. ∀e@i.first(e) ⇒ (r when e) = false₂
. . | 19. ∀e@i.¬(r after e) = (r when e) ⇒ kind(e) = k
. . | 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v) ∨₂ r)
. . |     ∈ Dsys
. . | 21. vartype(i;r) ⊆r 𝔹
. . | 22. vartype(i;x) ⊆r A
. . | 23. ∀e:E. loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . | 24. ∀e:E.
. . |     loc(e) = i
. . |     ⇒
. . |     kind(e) = k
. . |     ⇒
. . |     (r after e) = (P((x when e),val(e)) ∨₂ (r when e))
. . | ⊢  ∀e@i.kind(e) = k ⇒ (valtype(e) ⊆r T) & (vartype(i;x) ⊆r A)
. . |   by Analyze 0 THEN Unfold `alle-at` 0 THEN Trivial
. . |
. . .\
. . . 1. T : Type
. . . 2. A : Type
. . . 3. P : A→T→𝔹
. . . 4. k : Knd
. . . 5. i : Id
. . . 6. r : Id

. . . 7. x : Id
. . . 8. A
. . . 9. T
. . . 10. ¬x = r
. . . 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . 12. D' : Dsys
. . . 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . 14. w : World
. . . 15. p : FairFifo
. . . 16. PossibleWorld(D';w)
. . . 17. vartype(i;r) ⊆r 𝔹
. . . 18. ∀e@i.first(e) ⇒ (r when e) = false₂
. . . 19. ∀e@i.¬(r after e) = (r when e) ⇒ kind(e) = k
. . . 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v) ∨₂ r)
. . . |   ∈ Dsys
. . . 21. vartype(i;r) ⊆r 𝔹
. . . 22. vartype(i;x) ⊆r A
. . . 23. ∀e:E. loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
```

```
. . . 24. ∀e:E.
. . . |   loc(e) = i
. . . |   ⇒
. . . |   kind(e) = k
. . . |   ⇒
. . . |   (r after e) = (P((x when e),val(e)) ∨2 (r when e))
. . . ⊢  ∀e@i.first(e) ⇒ (r when e) = false2
. . . |  & ∀e'@i.(r after e')
. . . |          ⟺
. . . |          (∃e:E.
. . . |           (e <loc e') ∨ e = e'
. . . |           & kind(e) = k
. . . |           & P((x when e),val(e)))
. . . | by Analyze 0 THEN Try Trivial
. . . |
. . . 1. T : Type
. . . 2. A : Type
. . . 3. P : A→T→𝔹
. . . 4. k : Knd
. . . 5. i : Id
. . . 6. r : Id
. . . 7. x : Id
. . . 8. A
. . . 9. T
. . . 10. ¬x = r
. . . 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . 12. D' : Dsys
. . . 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . 14. w : World
. . . 15. p : FairFifo
. . . 16. PossibleWorld(D';w)
. . . 17. vartype(i;r) ⊆r 𝔹
. . . 18. ∀e@i.first(e) ⇒ (r when e) = false2
. . . 19. ∀e@i.¬(r after e) = (r when e) ⇒ kind(e) = k
. . . 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v) ∨2 r)
. . . |   ∈ Dsys
. . . 21. vartype(i;r) ⊆r 𝔹
. . . 22. vartype(i;x) ⊆r A
. . . 23. ∀e:E. loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . 24. ∀e:E.
. . . |   loc(e) = i
. . . |   ⇒
. . . |   kind(e) = k
. . . |   ⇒
. . . |   (r after e) = (P((x when e),val(e)) ∨2 (r when e))
. . . ⊢  ∀e'@i.(r after e')
. . . |        ⟺
. . . |        (∃e:E.
. . . |         (e <loc e') ∨ e = e'
. . . |         & kind(e) = k
. . . |         & P((x when e),val(e)))
. . . | by Unfold `and` 0 THEN Fold `cand` 0
. . . |    THEN
. . . |    BackThru
. . . |    Thm* ∀es:ES, i:Id, P:({e:E| loc(e) = i ∈ Id }→Prop).
```

```
. . . |         ∀e@i.P(e)
. . . |         ⟺
. . . |         ∀e@i.first(e) ⇒ P(e)
. . . |         & ∀e@i.¬first(e) ⇒ P(pred(e)) ⇒ P(e)
. . . |    {alle_DASH_at_DASH_iff}
. . . |    THENA
. . . |    (MaAuto THEN SplitOrHyps THEN MaAuto)
. . . |
. . . 1. T : Type
. . . 2. A : Type
. . . 3. P : A→T→𝔹
. . . 4. k : Knd
. . . 5. i : Id
. . . 6. r : Id
. . . 7. x : Id
. . . 8. A
. . . 9. T
. . . 10. ¬x = r
. . . 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . 12. D' : Dsys
. . . 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . 14. w : World
. . . 15. p : FairFifo
. . . 16. PossibleWorld(D';w)
. . . 17. vartype(i;r) ⊆r 𝔹
. . . 18. ∀e@i.first(e) ⇒ (r when e) = false₂
. . . 19. ∀e@i.¬(r after e) = (r when e) ⇒ kind(e) = k
. . . 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v) ∨₂ r)
. . . |   ∈ Dsys
. . . 21. vartype(i;r) ⊆r 𝔹
. . . 22. vartype(i;x) ⊆r A
. . . 23. ∀e:E. loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . 24. ∀e:E.
. . . |   loc(e) = i
. . . |   ⇒
. . . |   kind(e) = k
. . . |   ⇒
. . . |   (r after e) = (P((x when e),val(e)) ∨₂ (r when e))
. . . ⊢  ∀e'@i.first(e')
. . . |        ⇒
. . . |  ((r after e')
. . . |   ⟺
. . . |   (∃e@0:E.
. . . |    (e@0 <loc e') ∨ e@0 = e'
. . . |    & kind(e@0) = k
. . . |    & P((x when e@0),val(e@0))))
. . . |  & ∀e'@i.¬first(e')
. . . |          ⇒
. . . |    ((r after pred(e'))
. . . |     ⟺
. . . |     (∃e@0:E.
. . . |      (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . |      & kind(e@0) = k
. . . |      & P((x when e@0),val(e@0))))
. . . |    ⇒
```

```
. . . |    ((r after e')
. . . |     ⟺
. . . |     (∃e@0:E.
. . . |      (e@0 <loc e') ∨ e@0 = e'
. . . |      & kind(e@0) = k
. . . |      & P((x when e@0),val(e@0))))  by Analyze 0
. . . |
. . . |\
. . . | 1. T : Type
. . . | 2. A : Type
. . . | 3. P : A→T→𝔹
. . . | 4. k : Knd
. . . | 5. i : Id
. . . | 6. r : Id
. . . | 7. x : Id
. . . | 8. A
. . . | 9. T
. . . | 10. ¬x = r
. . . | 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . | 12. D' : Dsys
. . . | 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . | 14. w : World
. . . | 15. p : FairFifo
. . . | 16. PossibleWorld(D';w)
. . . | 17. vartype(i;r) ⊆r 𝔹
. . . | 18. ∀e@i.first(e) ⇒ (r when e) = false2
. . . | 19. ∀e@i.¬(r after e) = (r when e) ⇒ kind(e) = k
. . . | 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v) ∨2 r)
. . . | |   ∈ Dsys
. . . | 21. vartype(i;r) ⊆r 𝔹
. . . | 22. vartype(i;x) ⊆r A
. . . | 23. ∀e:E. loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . | 24. ∀e:E.
. . . | |   loc(e) = i
. . . | |   ⇒
. . . | |   kind(e) = k
. . . | |   ⇒
. . . | |   (r after e) = (P((x when e),val(e)) ∨2 (r when e))
. . . | ⊢  ∀e'@i.first(e')
. . . | |        ⇒
. . . | |  ((r after e')
. . . | |   ⟺
. . . | |   (∃e@0:E.
. . . | |    (e@0 <loc e') ∨ e@0 = e'
. . . | |    & kind(e@0) = k
. . . | |    & P((x when e@0),val(e@0))))
. . . | | by All (Unfold `alle-at`) THEN MaAuto
. . . | |
. . . | |\
. . . | | 1. T : Type
. . . | | 2. A : Type
. . . | | 3. P : A→T→𝔹
. . . | | 4. k : Knd
. . . | | 5. i : Id
. . . | | 6. r : Id
```

```
. . . | | 7. x : Id
. . . | | 8. A
. . . | | 9. T
. . . | | 10. ¬x = r
. . . | | 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . | | 12. D' : Dsys
. . . | | 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . | | 14. w : World
. . . | | 15. p : FairFifo
. . . | | 16. PossibleWorld(D';w)
. . . | | 17. vartype(i;r) ⊆r 𝔹
. . . | | 18. ∀e:E. loc(e) = i ⇒ first(e) ⇒ (r when e) = false_2
. . . | | 19. ∀e:E.
. . . | | |   loc(e) = i ⇒ ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . | | 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v) ∨_2 r)
. . . | | |   ∈ Dsys
. . . | | 21. vartype(i;r) ⊆r 𝔹
. . . | | 22. vartype(i;x) ⊆r A
. . . | | 23. ∀e:E. loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . | | 24. ∀e:E.
. . . | | |   loc(e) = i
. . . | | |   ⇒
. . . | | |   kind(e) = k
. . . | | |   ⇒
. . . | | |   (r after e) = (P((x when e),val(e)) ∨_2 (r when e))
. . . | | 25. e' : E
. . . | | 26. loc(e') = i
. . . | | 27. first(e')
. . . | | 28. (r after e')
. . . | | ⊢  ∃e@0:E.
. . . | | |  (e@0 <loc e') ∨ e@0 = e'
. . . | | |  & kind(e@0) = k
. . . | | |  & P((x when e@0),val(e@0))  by Assert (kind(e') = k)
. . . | | |
. . . | | |\
. . . | | | .....assertion .....
. . . | | | 1. T : Type
. . . | | | 2. A : Type
. . . | | | 3. P : A→T→𝔹
. . . | | | 4. k : Knd
. . . | | | 5. i : Id
. . . | | | 6. r : Id
. . . | | | 7. x : Id
. . . | | | 8. A
. . . | | | 9. T
. . . | | | 10. ¬x = r
. . . | | | 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . | | | 12. D' : Dsys
. . . | | | 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . | | | 14. w : World
. . . | | | 15. p : FairFifo
. . . | | | 16. PossibleWorld(D';w)
. . . | | | 17. vartype(i;r) ⊆r 𝔹
. . . | | | 18. ∀e:E. loc(e) = i ⇒ first(e) ⇒ (r when e) = false_2
. . . | | | 19. ∀e:E.
```

```
. . . | | | |   loc(e) = i ⇒ ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . | | | 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v) ∨₂ r)
. . . | | | |   ∈ Dsys
. . . | | | 21. vartype(i;r) ⊆r 𝔹
. . . | | | 22. vartype(i;x) ⊆r A
. . . | | | 23. ∀e:E. loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . | | | 24. ∀e:E.
. . . | | | |   loc(e) = i
. . . | | | |   ⇒
. . . | | | |   kind(e) = k
. . . | | | |   ⇒
. . . | | | |   (r after e) = (P((x when e),val(e)) ∨₂ (r when e))
. . . | | | 25. e' : E
. . . | | | 26. loc(e') = i
. . . | | | 27. first(e')
. . . | | | 28. (r after e')
. . . | | | ⊢  kind(e') = k  by BackThruSomeHyp
. . . | | | |
. . . | | | 1. T : Type
. . . | | | 2. A : Type
. . . | | | 3. P : A→T→𝔹
. . . | | | 4. k : Knd
. . . | | | 5. i : Id
. . . | | | 6. r : Id
. . . | | | 7. x : Id
. . . | | | 8. A
. . . | | | 9. T
. . . | | | 10. ¬x = r
. . . | | | 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . | | | 12. D' : Dsys
. . . | | | 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . | | | 14. w : World
. . . | | | 15. p : FairFifo
. . . | | | 16. PossibleWorld(D';w)
. . . | | | 17. vartype(i;r) ⊆r 𝔹
. . . | | | 18. ∀e:E. loc(e) = i ⇒ first(e) ⇒ (r when e) = false₂
. . . | | | 19. ∀e:E.
. . . | | | |   loc(e) = i ⇒ ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . | | | 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v) ∨₂ r)
. . . | | | |   ∈ Dsys
. . . | | | 21. vartype(i;r) ⊆r 𝔹
. . . | | | 22. vartype(i;x) ⊆r A
. . . | | | 23. ∀e:E. loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . | | | 24. ∀e:E.
. . . | | | |   loc(e) = i
. . . | | | |   ⇒
. . . | | | |   kind(e) = k
. . . | | | |   ⇒
. . . | | | |   (r after e) = (P((x when e),val(e)) ∨₂ (r when e))
. . . | | | 25. e' : E
. . . | | | 26. loc(e') = i
. . . | | | 27. first(e')
. . . | | | 28. (r after e')
. . . | | | ⊢  ¬(r after e') = (r when e')
. . . | | | | by MoveToConcl -1 THEN AutoBoolCase (r after e')
```

```
. . . | | | |    THEN
. . . | | | |    MaAuto
. . . | | | |    THEN
. . . | | | |    AllHyps (λh.InstHyp [e'] h THENA Complete Auto)
. . . | | | |
. . . | | | 1. T : Type
. . . | | | 2. A : Type
. . . | | | 3. P : A→T→𝔹
. . . | | | 4. k : Knd
. . . | | | 5. i : Id
. . . | | | 6. r : Id
. . . | | | 7. x : Id
. . . | | | 8. A
. . . | | | 9. T
. . . | | | 10. ¬x = r
. . . | | | 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . | | | 12. D' : Dsys
. . . | | | 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . | | | 14. w : World
. . . | | | 15. p : FairFifo
. . . | | | 16. PossibleWorld(D';w)
. . . | | | 17. vartype(i;r) ⊆r 𝔹
. . . | | | 18. ∀e:E. loc(e) = i ⇒ first(e) ⇒ (r when e) = false₂
. . . | | | 19. ∀e:E.
. . . | | | |   loc(e) = i ⇒ ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . | | | 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v) ∨₂ r)
. . . | | | |   ∈ Dsys
. . . | | | 21. vartype(i;r) ⊆r 𝔹
. . . | | | 22. vartype(i;x) ⊆r A
. . . | | | 23. ∀e:E. loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . | | | 24. ∀e:E.
. . . | | | |   loc(e) = i
. . . | | | |   ⇒
. . . | | | |   kind(e) = k
. . . | | | |   ⇒
. . . | | | |   (r after e) = (P((x when e),val(e)) ∨₂ (r when e))
. . . | | | 25. e' : E
. . . | | | 26. loc(e') = i
. . . | | | 27. first(e')
. . . | | | 28. (r after e')
. . . | | | 29. True
. . . | | | 30. (r when e') = false₂
. . . | | | ⊢  ¬true₂ = (r when e')  by Analyze 0
. . . | | | |
. . . | | | 1. T : Type
. . . | | | 2. A : Type
. . . | | | 3. P : A→T→𝔹
. . . | | | 4. k : Knd
. . . | | | 5. i : Id
. . . | | | 6. r : Id
. . . | | | 7. x : Id
. . . | | | 8. A
. . . | | | 9. T
. . . | | | 10. ¬x = r
. . . | | | 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
```

```
. . . | | | 12. D' : Dsys
. . . | | | 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . | | | 14. w : World
. . . | | | 15. p : FairFifo
. . . | | | 16. PossibleWorld(D';w)
. . . | | | 17. vartype(i;r) ⊆r 𝔹
. . . | | | 18. ∀e:E. loc(e) = i ⇒ first(e) ⇒ (r when e) = false_2
. . . | | | 19. ∀e:E.
. . . | | | |   loc(e) = i ⇒ ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . | | | 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v) ∨_2 r)
. . . | | | |   ∈ Dsys
. . . | | | 21. vartype(i;r) ⊆r 𝔹
. . . | | | 22. vartype(i;x) ⊆r A
. . . | | | 23. ∀e:E. loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . | | | 24. ∀e:E.
. . . | | | |   loc(e) = i
. . . | | | |   ⇒
. . . | | | |   kind(e) = k
. . . | | | |   ⇒
. . . | | | |   (r after e) = (P((x when e),val(e)) ∨_2 (r when e))
. . . | | | 25. e' : E
. . . | | | 26. loc(e') = i
. . . | | | 27. first(e')
. . . | | | 28. (r after e')
. . . | | | 29. True
. . . | | | 30. (r when e') = false_2
. . . | | | 31. true_2 = (r when e')
. . . | | | ⊢  False  by Assert (true_2 = false_2)
. . . | | | |
. . . | | | 1. T : Type
. . . | | | 2. A : Type
. . . | | | 3. P : A→T→𝔹
. . . | | | 4. k : Knd
. . . | | | 5. i : Id
. . . | | | 6. r : Id
. . . | | | 7. x : Id
. . . | | | 8. A
. . . | | | 9. T
. . . | | | 10. ¬x = r
. . . | | | 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . | | | 12. D' : Dsys
. . . | | | 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . | | | 14. w : World
. . . | | | 15. p : FairFifo
. . . | | | 16. PossibleWorld(D';w)
. . . | | | 17. vartype(i;r) ⊆r 𝔹
. . . | | | 18. ∀e:E. loc(e) = i ⇒ first(e) ⇒ (r when e) = false_2
. . . | | | 19. ∀e:E.
. . . | | |     loc(e) = i ⇒ ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . | | | 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v) ∨_2 r)
. . . | | |     ∈ Dsys
. . . | | | 21. vartype(i;r) ⊆r 𝔹

. . . | | | 22. vartype(i;x) ⊆r A
. . . | | | 23. ∀e:E. loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . | | | 24. ∀e:E.
```

```
. . . | | |     loc(e) = i
. . . | | |     ⇒
. . . | | |     kind(e) = k
. . . | | |     ⇒
. . . | | |     (r after e) = (P((x when e),val(e)) ∨2 (r when e))
. . . | | | 25. e' : E
. . . | | | 26. loc(e') = i
. . . | | | 27. first(e')
. . . | | | 28. (r after e')
. . . | | | 29. True
. . . | | | 30. (r when e') = false2
. . . | | | 31. true2 = (r when e')
. . . | | | 32. true2 = false2
. . . | | | ⊢  False  by ObviousFrom [-1]
. . . | | |
. . . | | .\
. . . | | . 1. T : Type
. . . | | . 2. A : Type
. . . | | . 3. P : A→T→𝔹
. . . | | . 4. k : Knd
. . . | | . 5. i : Id
. . . | | . 6. r : Id
. . . | | . 7. x : Id
. . . | | . 8. A
. . . | | . 9. T
. . . | | . 10. ¬x = r
. . . | | . 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . | | . 12. D' : Dsys
. . . | | . 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . | | . 14. w : World
. . . | | . 15. p : FairFifo
. . . | | . 16. PossibleWorld(D';w)
. . . | | . 17. vartype(i;r) ⊆r 𝔹
. . . | | . 18. ∀e:E. loc(e) = i ⇒ first(e) ⇒ (r when e) = false2
. . . | | . 19. ∀e:E.
. . . | | . |   loc(e) = i ⇒ ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . | | . 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v) ∨2 r)
. . . | | . |   ∈ Dsys
. . . | | . 21. vartype(i;r) ⊆r 𝔹
. . . | | . 22. vartype(i;x) ⊆r A
. . . | | . 23. ∀e:E. loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . | | . 24. ∀e:E.
. . . | | . |   loc(e) = i
. . . | | . |   ⇒
. . . | | . |   kind(e) = k
. . . | | . |   ⇒
. . . | | . |   (r after e) = (P((x when e),val(e)) ∨2 (r when e))
. . . | | . 25. e' : E
. . . | | . 26. loc(e') = i
. . . | | . 27. first(e')
. . . | | . 28. (r after e')
. . . | | . 29. kind(e') = k
. . . | | . ⊢  ∃e@0:E.
. . . | | . |  (e@0 <loc e') ∨ e@0 = e'
. . . | | . |  & kind(e@0) = k
```

```
. . . | | . |  & P((x when e@0),val(e@0))
. . . | | . | by AllHyps (λh.InstHyp [e'] h THENA Complete Auto)
. . . | | . |    THEN
. . . | | . |    InstConcl [e']
. . . | | . |    THEN
. . . | | . |    MaAuto
. . . | | . |
. . . | | . |\
. . . | | . | 1. T : Type
. . . | | . | 2. A : Type
. . . | | . | 3. P : A→T→𝔹
. . . | | . | 4. k : Knd
. . . | | . | 5. i : Id
. . . | | . | 6. r : Id
. . . | | . | 7. x : Id
. . . | | . | 8. A
. . . | | . | 9. T
. . . | | . | 10. ¬x = r
. . . | | . | 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . | | . | 12. D' : Dsys
. . . | | . | 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . | | . | 14. w : World
. . . | | . | 15. p : FairFifo
. . . | | . | 16. PossibleWorld(D';w)
. . . | | . | 17. vartype(i;r) ⊆r 𝔹
. . . | | . | 18. ∀e:E. loc(e) = i ⇒ first(e) ⇒ (r when e) = false₂
. . . | | . | 19. ∀e:E.
. . . | | . |     loc(e) = i
. . . | | . |     ⇒
. . . | | . |     ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . | | . | 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v)
. . . | | . |     ∨₂ r)
. . . | | . |     ∈ Dsys
. . . | | . | 21. vartype(i;r) ⊆r 𝔹
. . . | | . | 22. vartype(i;x) ⊆r A
. . . | | . | 23. ∀e:E. loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . | | . | 24. ∀e:E.
. . . | | . |     loc(e) = i
. . . | | . |     ⇒
. . . | | . |     kind(e) = k
. . . | | . |     ⇒
. . . | | . |     (r after e) = (P((x when e),val(e)) ∨₂ (r when e))
. . . | | . | 25. e' : E
. . . | | . | 26. loc(e') = i
. . . | | . | 27. first(e')
. . . | | . | 28. (r after e')
. . . | | . | 29. kind(e') = k
. . . | | . | 30. (r after e')
. . . | | . |     =
. . . | | . |     (P((x when e'),val(e')) ∨₂ (r when e'))
. . . | | . | 31. valtype(e') ⊆r T
. . . | | . | 32. (r when e') = false₂
. . . | | . | ⊢  (e' <loc e') ∨ e' = e'  by OrRight
. . . | | . |
. . . | | . .\
```

```
. . . | | . . 1. T : Type
. . . | | . . 2. A : Type
. . . | | . . 3. P : A→T→𝔹
. . . | | . . 4. k : Knd
. . . | | . . 5. i : Id
. . . | | . . 6. r : Id
. . . | | . . 7. x : Id
. . . | | . . 8. A
. . . | | . . 9. T
. . . | | . . 10. ¬x = r
. . . | | . . 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . | | . . 12. D' : Dsys
. . . | | . . 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . | | . . 14. w : World
. . . | | . . 15. p : FairFifo
. . . | | . . 16. PossibleWorld(D';w)
. . . | | . . 17. vartype(i;r) ⊆r 𝔹
. . . | | . . 18. ∀e:E. loc(e) = i ⇒ first(e) ⇒ (r when e) = false₂
. . . | | . . 19. ∀e:E.
. . . | | . . |   loc(e) = i
. . . | | . . |   ⇒
. . . | | . . |   ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . | | . . 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v)
. . . | | . . |   ∨₂ r)
. . . | | . . |   ∈ Dsys
. . . | | . . 21. vartype(i;r) ⊆r 𝔹
. . . | | . . 22. vartype(i;x) ⊆r A
. . . | | . . 23. ∀e:E. loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . | | . . 24. ∀e:E.
. . . | | . . |   loc(e) = i
. . . | | . . |   ⇒
. . . | | . . |   kind(e) = k
. . . | | . . |   ⇒
. . . | | . . |   (r after e) = (P((x when e),val(e)) ∨₂ (r when e))
. . . | | . . 25. e' : E
. . . | | . . 26. loc(e') = i
. . . | | . . 27. first(e')
. . . | | . . 28. (r after e')
. . . | | . . 29. kind(e') = k
. . . | | . . 30. (r after e')
. . . | | . . |   =
. . . | | . . |   (P((x when e'),val(e')) ∨₂ (r when e'))
. . . | | . . 31. valtype(e') ⊆r T
. . . | | . . 32. (r when e') = false₂
. . . | | . . ⊢  P((x when e'),val(e'))
. . . | | . . | by HypSubst' -1 -3 THEN MoveToConcl -3
. . . | | . . |    THEN
. . . | | . . |    MoveToConcl -4
. . . | | . . |    THEN
. . . | | . . |    AutoBoolCase (r after e')
. . . | | . . |    THEN
. . . | | . . |    AutoBoolCase (P((x when e'),val(e')))
. . . | | . . |    THEN
. . . | | . . |    MaAuto
. . . | | . . |
```

```
. . . | | . . 1. T : Type
. . . | | . . 2. A : Type
. . . | | . . 3. P : A→T→𝔹
. . . | | . . 4. k : Knd
. . . | | . . 5. i : Id
. . . | | . . 6. r : Id
. . . | | . . 7. x : Id
. . . | | . . 8. A
. . . | | . . 9. T
. . . | | . . 10. ¬x = r
. . . | | . . 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . | | . . 12. D' : Dsys
. . . | | . . 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . | | . . 14. w : World
. . . | | . . 15. p : FairFifo
. . . | | . . 16. PossibleWorld(D';w)
. . . | | . . 17. vartype(i;r) ⊆r 𝔹
. . . | | . . 18. ∀e:E. loc(e) = i ⇒ first(e) ⇒ (r when e) = false_2
. . . | | . . 19. ∀e:E.
. . . | | . .     loc(e) = i
. . . | | . .     ⇒
. . . | | . .     ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . | | . . 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v)
. . . | | . .     ∨_2 r)
. . . | | . .     ∈ Dsys
. . . | | . . 21. vartype(i;r) ⊆r 𝔹
. . . | | . . 22. vartype(i;x) ⊆r A
. . . | | . . 23. ∀e:E. loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . | | . . 24. ∀e:E.
. . . | | . .     loc(e) = i
. . . | | . .     ⇒
. . . | | . .     kind(e) = k
. . . | | . .     ⇒
. . . | | . .     (r after e) = (P((x when e),val(e)) ∨_2 (r when e))
. . . | | . . 25. e' : E
. . . | | . . 26. loc(e') = i
. . . | | . . 27. first(e')
. . . | | . . 28. kind(e') = k
. . . | | . . 29. valtype(e') ⊆r T
. . . | | . . 30. (r when e') = false_2
. . . | | . . 31. (r after e')
. . . | | . . 32. ¬P((x when e'),val(e'))
. . . | | . . 33. True
. . . | | . . 34. true_2 = false_2
. . . | | . . ⊢  False  by ObviousFrom [-1]
. . . | |
. . . | .\
. . . | . 1. T : Type
. . . | . 2. A : Type
. . . | . 3. P : A→T→𝔹
. . . | . 4. k : Knd
. . . | . 5. i : Id
. . . | . 6. r : Id
. . . | . 7. x : Id
. . . | . 8. A
```

```
. . . | . 9. T
. . . | . 10. ¬x = r
. . . | . 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . | . 12. D' : Dsys
. . . | . 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . | . 14. w : World
. . . | . 15. p : FairFifo
. . . | . 16. PossibleWorld(D';w)
. . . | . 17. vartype(i;r) ⊆r 𝔹
. . . | . 18. ∀e:E. loc(e) = i ⇒ first(e) ⇒ (r when e) = false2
. . . | . 19. ∀e:E.
. . . | . |   loc(e) = i ⇒ ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . | . 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v) ∨2 r)
. . . | . |   ∈ Dsys
. . . | . 21. vartype(i;r) ⊆r 𝔹
. . . | . 22. vartype(i;x) ⊆r A
. . . | . 23. ∀e:E. loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . | . 24. ∀e:E.
. . . | . |   loc(e) = i
. . . | . |   ⇒
. . . | . |   kind(e) = k
. . . | . |   ⇒
. . . | . |   (r after e) = (P((x when e),val(e)) ∨2 (r when e))
. . . | . 25. e' : E
. . . | . 26. loc(e') = i
. . . | . 27. first(e')
. . . | . 28. ∃e@0:E.
. . . | . |   (e@0 <loc e') ∨ e@0 = e'
. . . | . |   & kind(e@0) = k
. . . | . |   & P((x when e@0),val(e@0))
. . . | . ⊢  (r after e')  by ExRepD
. . . | . |
. . . | . 1. T : Type
. . . | . 2. A : Type
. . . | . 3. P : A→T→𝔹
. . . | . 4. k : Knd
. . . | . 5. i : Id
. . . | . 6. r : Id
. . . | . 7. x : Id
. . . | . 8. A
. . . | . 9. T
. . . | . 10. ¬x = r
. . . | . 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . | . 12. D' : Dsys
. . . | . 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . | . 14. w : World
. . . | . 15. p : FairFifo
. . . | . 16. PossibleWorld(D';w)
. . . | . 17. vartype(i;r) ⊆r 𝔹
. . . | . 18. ∀e:E. loc(e) = i ⇒ first(e) ⇒ (r when e) = false2
. . . | . 19. ∀e:E.
. . . | . |   loc(e) = i ⇒ ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . | . 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v) ∨2 r)
. . . | . |   ∈ Dsys
. . . | . 21. vartype(i;r) ⊆r 𝔹
```

```
. . . | . 22. vartype(i;x) ⊆r A
. . . | . 23. ∀e:E. loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . | . 24. ∀e:E.
. . . | . |   loc(e) = i
. . . | . |   ⇒
. . . | . |   kind(e) = k
. . . | . |   ⇒
. . . | . |   (r after e) = (P((x when e),val(e)) ∨2 (r when e))
. . . | . 25. e' : E
. . . | . 26. loc(e') = i
. . . | . 27. first(e')
. . . | . 28. e@0 : E
. . . | . 29. (e@0 <loc e') ∨ e@0 = e'
. . . | . 30. kind(e@0) = k
. . . | . 31. P((x when e@0),val(e@0))
. . . | . ⊢  (r after e')  by Analyze -3
. . . | . |
. . . | . |\
. . . | . | 1. T : Type
. . . | . | 2. A : Type
. . . | . | 3. P : A→T→𝔹
. . . | . | 4. k : Knd
. . . | . | 5. i : Id
. . . | . | 6. r : Id
. . . | . | 7. x : Id
. . . | . | 8. A
. . . | . | 9. T
. . . | . | 10. ¬x = r
. . . | . | 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . | . | 12. D' : Dsys
. . . | . | 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . | . | 14. w : World
. . . | . | 15. p : FairFifo
. . . | . | 16. PossibleWorld(D';w)
. . . | . | 17. vartype(i;r) ⊆r 𝔹
. . . | . | 18. ∀e:E. loc(e) = i ⇒ first(e) ⇒ (r when e) = false2
. . . | . | 19. ∀e:E.
. . . | . |     loc(e) = i ⇒ ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . | . | 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v) ∨2 r)
. . . | . |     ∈ Dsys
. . . | . | 21. vartype(i;r) ⊆r 𝔹
. . . | . | 22. vartype(i;x) ⊆r A
. . . | . | 23. ∀e:E. loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . | . | 24. ∀e:E.
. . . | . |     loc(e) = i
. . . | . |     ⇒
. . . | . |     kind(e) = k
. . . | . |     ⇒
. . . | . |     (r after e) = (P((x when e),val(e)) ∨2 (r when e))
. . . | . | 25. e' : E
. . . | . | 26. loc(e') = i
. . . | . | 27. first(e')
. . . | . | 28. e@0 : E
. . . | . | 29. (e@0 <loc e')
. . . | . | 30. kind(e@0) = k
```

```
. . . | . | 31. P((x when e@0),val(e@0))
. . . | . | ⊢  (r after e')
. . . | . |   by InstESAxiom ES(w) [e'] 4 THEN ThinTrivial
. . . | . |      THEN
. . . | . |      InstHyp [e@0] -1
. . . | . |
. . . | . .\
. . . | . . 1. T : Type
. . . | . . 2. A : Type
. . . | . . 3. P : A→T→𝔹
. . . | . . 4. k : Knd
. . . | . . 5. i : Id
. . . | . . 6. r : Id
. . . | . . 7. x : Id
. . . | . . 8. A
. . . | . . 9. T
. . . | . . 10. ¬x = r
. . . | . . 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . | . . 12. D' : Dsys
. . . | . . 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . | . . 14. w : World
. . . | . . 15. p : FairFifo
. . . | . . 16. PossibleWorld(D';w)
. . . | . . 17. vartype(i;r) ⊆r 𝔹
. . . | . . 18. ∀e:E. loc(e) = i ⇒ first(e) ⇒ (r when e) = false₂
. . . | . . 19. ∀e:E.
. . . | . . |   loc(e) = i ⇒ ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . | . . 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v) ∨₂ r)
. . . | . . |   ∈ Dsys
. . . | . . 21. vartype(i;r) ⊆r 𝔹
. . . | . . 22. vartype(i;x) ⊆r A
. . . | . . 23. ∀e:E. loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . | . . 24. ∀e:E.
. . . | . . |   loc(e) = i
. . . | . . |   ⇒
. . . | . . |   kind(e) = k
. . . | . . |   ⇒
. . . | . . |   (r after e) = (P((x when e),val(e)) ∨₂ (r when e))
. . . | . . 25. e' : E
. . . | . . 26. loc(e') = i
. . . | . . 27. first(e')
. . . | . . 28. e@0 : E
. . . | . . 29. e@0 = e'
. . . | . . 30. kind(e@0) = k
. . . | . . 31. P((x when e@0),val(e@0))
. . . | . . ⊢  (r after e')  by RW (SweepDnC (HypC -8)) 0
. . . | . . |
. . . | . . |\
. . . | . . | .....rewrite subgoal .....
. . . | . . | 1. T : Type
. . . | . . | 2. A : Type
. . . | . . | 3. P : A→T→𝔹
. . . | . . | 4. k : Knd
. . . | . . | 5. i : Id
. . . | . . | 6. r : Id
```

```
. . . | . . | 7. x : Id
. . . | . . | 8. A
. . . | . . | 9. T
. . . | . . | 10. ¬x = r
. . . | . . | 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . | . . | 12. D' : Dsys
. . . | . . | 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . | . . | 14. w : World
. . . | . . | 15. p : FairFifo
. . . | . . | 16. PossibleWorld(D';w)
. . . | . . | 17. vartype(i;r) ⊆r 𝔹
. . . | . . | 18. ∀e:E. loc(e) = i ⇒ first(e) ⇒ (r when e) = false₂
. . . | . . | 19. ∀e:E.
. . . | . . |     loc(e) = i
. . . | . . |     ⇒
. . . | . . |     ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . | . . | 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v)
. . . | . . |     ∨₂ r)
. . . | . . |     ∈ Dsys
. . . | . . | 21. vartype(i;r) ⊆r 𝔹
. . . | . . | 22. vartype(i;x) ⊆r A
. . . | . . | 23. ∀e:E. loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . | . . | 24. ∀e:E.
. . . | . . |     loc(e) = i
. . . | . . |     ⇒
. . . | . . |     kind(e) = k
. . . | . . |     ⇒
. . . | . . |     (r after e) = (P((x when e),val(e)) ∨₂ (r when e))
. . . | . . | 25. e' : E
. . . | . . | 26. loc(e') = i
. . . | . . | 27. first(e')
. . . | . . | 28. e@0 : E
. . . | . . | 29. e@0 = e'
. . . | . . | 30. kind(e@0) = k
. . . | . . | 31. P((x when e@0),val(e@0))
. . . | . . | ⊢  kind(e') = k  by HypSubst -3 -2
. . . | . . |
. . . | . . .\
. . . | . . . 1. T : Type
. . . | . . . 2. A : Type
. . . | . . . 3. P : A→T→𝔹
. . . | . . . 4. k : Knd
. . . | . . . 5. i : Id
. . . | . . . 6. r : Id
. . . | . . . 7. x : Id
. . . | . . . 8. A
. . . | . . . 9. T
. . . | . . . 10. ¬x = r
. . . | . . . 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . | . . . 12. D' : Dsys
. . . | . . . 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . | . . . 14. w : World
. . . | . . . 15. p : FairFifo
. . . | . . . 16. PossibleWorld(D';w)
. . . | . . . 17. vartype(i;r) ⊆r 𝔹
```

```
. . . | . . . 18. ∀e:E. loc(e) = i ⇒ first(e) ⇒ (r when e) = false2
. . . | . . . 19. ∀e:E.
. . . | . . . |   loc(e) = i
. . . | . . . |   ⇒
. . . | . . . |   ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . | . . . 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v)
. . . | . . . |   ∨2 r)
. . . | . . . |   ∈ Dsys
. . . | . . . 21. vartype(i;r) ⊆r 𝔹
. . . | . . . 22. vartype(i;x) ⊆r A
. . . | . . . 23. ∀e:E. loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . | . . . 24. ∀e:E.
. . . | . . . |   loc(e) = i
. . . | . . . |   ⇒
. . . | . . . |   kind(e) = k
. . . | . . . |   ⇒
. . . | . . . |   (r after e) = (P((x when e),val(e)) ∨2 (r when e))
. . . | . . . 25. e' : E
. . . | . . . 26. loc(e') = i
. . . | . . . 27. first(e')
. . . | . . . 28. e@0 : E
. . . | . . . 29. e@0 = e'
. . . | . . . 30. kind(e@0) = k
. . . | . . . 31. P((x when e@0),val(e@0))
. . . | . . . ⊢  P((x when e'),val(e')) ∨2 (r when e')
. . . | . . . | by RW assert_pushdownC 0 THEN MaAuto
. . . | . . . |
. . . | . . . |\
. . . | . . . | .....wf .....
. . . | . . . | 1. T : Type
. . . | . . . | 2. A : Type
. . . | . . . | 3. P : A→T→𝔹
. . . | . . . | 4. k : Knd
. . . | . . . | 5. i : Id
. . . | . . . | 6. r : Id
. . . | . . . | 7. x : Id
. . . | . . . | 8. A
. . . | . . . | 9. T
. . . | . . . | 10. ¬x = r
. . . | . . . | 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . | . . . | 12. D' : Dsys
. . . | . . . | 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . | . . . | 14. w : World
. . . | . . . | 15. p : FairFifo
. . . | . . . | 16. PossibleWorld(D';w)
. . . | . . . | 17. vartype(i;r) ⊆r 𝔹
. . . | . . . | 18. ∀e:E.
. . . | . . . |     loc(e) = i ⇒ first(e) ⇒ (r when e) = false2
. . . | . . . | 19. ∀e:E.
. . . | . . . |     loc(e) = i
. . . | . . . |     ⇒
. . . | . . . |     ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . | . . . | 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v)
. . . | . . . |     ∨2 r)
. . . | . . . |     ∈ Dsys
```

```
. . . | . . . | 21. vartype(i;r) ⊆r 𝔹
. . . | . . . | 22. vartype(i;x) ⊆r A
. . . | . . . | 23. ∀e:E.
. . . | . . . |     loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . | . . . | 24. ∀e:E.
. . . | . . . |     loc(e) = i
. . . | . . . |     ⇒
. . . | . . . |     kind(e) = k
. . . | . . . |     ⇒
. . . | . . . |     (r after e) = (P((x when e),val(e)) ∨₂ (r when e))
. . . | . . . | 25. e' : E
. . . | . . . | 26. loc(e') = i
. . . | . . . | 27. first(e')
. . . | . . . | 28. e@0 : E
. . . | . . . | 29. e@0 = e'
. . . | . . . | 30. kind(e@0) = k
. . . | . . . | 31. P((x when e@0),val(e@0))
. . . | . . . | ⊢  val(e') ∈ T
. . . | . . . |   by DoSubsume THEN BackThruSomeHyp THEN MaAuto
. . . | . . . |
. . . | . . . .\
. . . | . . . . 1. T : Type
. . . | . . . . 2. A : Type
. . . | . . . . 3. P : A→T→𝔹
. . . | . . . . 4. k : Knd
. . . | . . . . 5. i : Id
. . . | . . . . 6. r : Id
. . . | . . . . 7. x : Id
. . . | . . . . 8. A
. . . | . . . . 9. T
. . . | . . . . 10. ¬x = r
. . . | . . . . 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . | . . . . 12. D' : Dsys
. . . | . . . . 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . | . . . . 14. w : World
. . . | . . . . 15. p : FairFifo
. . . | . . . . 16. PossibleWorld(D';w)
. . . | . . . . 17. vartype(i;r) ⊆r 𝔹
. . . | . . . . 18. ∀e:E.
. . . | . . . . |   loc(e) = i ⇒ first(e) ⇒ (r when e) = false₂
. . . | . . . . 19. ∀e:E.
. . . | . . . . |   loc(e) = i
. . . | . . . . |   ⇒
. . . | . . . . |   ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . | . . . . 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v)
. . . | . . . . |   ∨₂ r)
. . . | . . . . |   ∈ Dsys
. . . | . . . . 21. vartype(i;r) ⊆r 𝔹
. . . | . . . . 22. vartype(i;x) ⊆r A
. . . | . . . . 23. ∀e:E.
. . . | . . . . |   loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . | . . . . 24. ∀e:E.
. . . | . . . . |   loc(e) = i
. . . | . . . . |   ⇒
. . . | . . . . |   kind(e) = k
```

```
. . . | . . . . |   ⇒
. . . | . . . . |   (r after e) = (P((x when e),val(e)) ∨₂ (r when e))
. . . | . . . . 25. e' : E
. . . | . . . . 26. loc(e') = i
. . . | . . . . 27. first(e')
. . . | . . . . 28. e@0 : E
. . . | . . . . 29. e@0 = e'
. . . | . . . . 30. kind(e@0) = k
. . . | . . . . 31. P((x when e@0),val(e@0))
. . . | . . . . ⊢  P((x when e'),val(e')) ∨ (r when e')
. . . | . . . . | by OrLeft THEN MaAuto
. . . | . . . . |
. . . | . . . . 1. T : Type
. . . | . . . . 2. A : Type
. . . | . . . . 3. P : A→T→𝔹
. . . | . . . . 4. k : Knd
. . . | . . . . 5. i : Id
. . . | . . . . 6. r : Id
. . . | . . . . 7. x : Id
. . . | . . . . 8. A
. . . | . . . . 9. T
. . . | . . . . 10. ¬x = r
. . . | . . . . 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . | . . . . 12. D' : Dsys
. . . | . . . . 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . | . . . . 14. w : World
. . . | . . . . 15. p : FairFifo
. . . | . . . . 16. PossibleWorld(D';w)
. . . | . . . . 17. vartype(i;r) ⊆r 𝔹
. . . | . . . . 18. ∀e:E.
. . . | . . . . |   loc(e) = i ⇒ first(e) ⇒ (r when e) = false₂
. . . | . . . . 19. ∀e:E.
. . . | . . . . |   loc(e) = i
. . . | . . . . |   ⇒
. . . | . . . . |   ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . | . . . . 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v)
. . . | . . . . |   ∨₂ r)
. . . | . . . . |   ∈ Dsys
. . . | . . . . 21. vartype(i;r) ⊆r 𝔹
. . . | . . . . 22. vartype(i;x) ⊆r A
. . . | . . . . 23. ∀e:E.
. . . | . . . . |   loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . | . . . . 24. ∀e:E.
. . . | . . . . |   loc(e) = i
. . . | . . . . |   ⇒
. . . | . . . . |   kind(e) = k
. . . | . . . . |   ⇒
. . . | . . . . |   (r after e) = (P((x when e),val(e)) ∨₂ (r when e))
. . . | . . . . 25. e' : E
. . . | . . . . 26. loc(e') = i
. . . | . . . . 27. first(e')
. . . | . . . . 28. e@0 : E
. . . | . . . . 29. e@0 = e'
. . . | . . . . 30. kind(e@0) = k
. . . | . . . . 31. P((x when e@0),val(e@0))
```

```
. . . | . . . . ⊢  P((x when e'),val(e'))
. . . | . . . . | by StrongRevHypSubst -3 0 THEN MaAuto
. . . | . . . . |
. . . | . . . . 1. T : Type
. . . | . . . . 2. A : Type
. . . | . . . . 3. P : A→T→𝔹
. . . | . . . . 4. k : Knd
. . . | . . . . 5. i : Id
. . . | . . . . 6. r : Id
. . . | . . . . 7. x : Id
. . . | . . . . 8. A
. . . | . . . . 9. T
. . . | . . . . 10. ¬x = r
. . . | . . . . 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . | . . . . 12. D' : Dsys
. . . | . . . . 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . | . . . . 14. w : World
. . . | . . . . 15. p : FairFifo
. . . | . . . . 16. PossibleWorld(D';w)
. . . | . . . . 17. vartype(i;r) ⊆r 𝔹
. . . | . . . . 18. ∀e:E.
. . . | . . . .     loc(e) = i ⇒ first(e) ⇒ (r when e) = false2
. . . | . . . . 19. ∀e:E.
. . . | . . . .     loc(e) = i
. . . | . . . .     ⇒
. . . | . . . .     ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . | . . . . 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v)
. . . | . . . .     ∨2 r)
. . . | . . . .     ∈ Dsys
. . . | . . . . 21. vartype(i;r) ⊆r 𝔹
. . . | . . . . 22. vartype(i;x) ⊆r A
. . . | . . . . 23. ∀e:E.
. . . | . . . .     loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . | . . . . 24. ∀e:E.
. . . | . . . .     loc(e) = i
. . . | . . . .     ⇒
. . . | . . . .     kind(e) = k
. . . | . . . .     ⇒
. . . | . . . .     (r after e) = (P((x when e),val(e)) ∨2 (r when e))
. . . | . . . . 25. e' : E
. . . | . . . . 26. loc(e') = i
. . . | . . . . 27. first(e')
. . . | . . . . 28. e@0 : E
. . . | . . . . 29. e@0 = e'
. . . | . . . . 30. kind(e@0) = k
. . . | . . . . 31. P((x when e@0),val(e@0))
. . . | . . . . 32. z : E
. . . | . . . . 33. z = e@0
. . . | . . . . ⊢  val(z) ∈ T
. . . | . . . .   by DoSubsume THEN HypSubst -1 0 THEN BackThruSomeHyp
. . . | . . . .     THEN
. . . | . . . .     MaAuto
. . . |
. . . .\
. . . . 1. T : Type
```

```
. . . . . 2. A : Type
. . . . . 3. P : A→T→𝔹
. . . . . 4. k : Knd
. . . . . 5. i : Id
. . . . . 6. r : Id
. . . . . 7. x : Id
. . . . . 8. A
. . . . . 9. T
. . . . . 10. ¬x = r
. . . . . 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . . . 12. D' : Dsys
. . . . . 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . . . 14. w : World
. . . . . 15. p : FairFifo
. . . . . 16. PossibleWorld(D';w)
. . . . . 17. vartype(i;r) ⊆r 𝔹
. . . . . 18. ∀e@i.first(e) ⇒ (r when e) = false₂
. . . . . 19. ∀e@i.¬(r after e) = (r when e) ⇒ kind(e) = k
. . . . . 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v) ∨₂ r)
. . . . . |    ∈ Dsys
. . . . . 21. vartype(i;r) ⊆r 𝔹
. . . . . 22. vartype(i;x) ⊆r A
. . . . . 23. ∀e:E. loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . . . 24. ∀e:E.
. . . . . |    loc(e) = i
. . . . . |    ⇒
. . . . . |    kind(e) = k
. . . . . |    ⇒
. . . . . |    (r after e) = (P((x when e),val(e)) ∨₂ (r when e))
. . . . . ⊢  ∀e'@i.¬first(e')
. . . . . |         ⇒
. . . . . |  ((r after pred(e'))
. . . . . |   ⟺
. . . . . |   (∃e@0:E.
. . . . . |    (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . . |    & kind(e@0) = k
. . . . . |    & P((x when e@0),val(e@0))))
. . . . . |  ⇒
. . . . . |  ((r after e')
. . . . . |   ⟺
. . . . . |   (∃e@0:E.
. . . . . |    (e@0 <loc e') ∨ e@0 = e'
. . . . . |    & kind(e@0) = k
. . . . . |    & P((x when e@0),val(e@0))))
. . . . . | by All (Unfold `alle-at`)
. . . . . |    THEN
. . . . . |    Repeat (Analyze 0 THENA Complete Auto)
. . . . . |    THEN
. . . . . |    Analyze 0
. . . . . |
. . . . . |\
. . . . . | 1. T : Type
. . . . . | 2. A : Type
. . . . . | 3. P : A→T→𝔹
. . . . . | 4. k : Knd
```

```
. . . . | 5. i : Id
. . . . | 6. r : Id
. . . . | 7. x : Id
. . . . | 8. A
. . . . | 9. T
. . . . | 10. ¬x = r
. . . . | 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . . | 12. D' : Dsys
. . . . | 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . . | 14. w : World
. . . . | 15. p : FairFifo
. . . . | 16. PossibleWorld(D';w)
. . . . | 17. vartype(i;r) ⊆r 𝔹
. . . . | 18. ∀e:E. loc(e) = i ⇒ first(e) ⇒ (r when e) = false₂
. . . . | 19. ∀e:E.
. . . . | |   loc(e) = i ⇒ ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . . | 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v) ∨₂ r)
. . . . | |   ∈ Dsys
. . . . | 21. vartype(i;r) ⊆r 𝔹
. . . . | 22. vartype(i;x) ⊆r A
. . . . | 23. ∀e:E. loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . . | 24. ∀e:E.
. . . . | |   loc(e) = i
. . . . | |   ⇒
. . . . | |   kind(e) = k
. . . . | |   ⇒
. . . . | |   (r after e) = (P((x when e),val(e)) ∨₂ (r when e))
. . . . | 25. e' : E
. . . . | 26. loc(e') = i
. . . . | 27. ¬first(e')
. . . . | 28. (r after pred(e'))
. . . . | |   ⟺
. . . . | |   (∃e@0:E.
. . . . | |    (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | |    & kind(e@0) = k
. . . . | |    & P((x when e@0),val(e@0)))
. . . . | ⊢  (r after e')
. . . . | |  ⟺
. . . . | |  (∃e@0:E.
. . . . | |   (e@0 <loc e') ∨ e@0 = e'
. . . . | |   & kind(e@0) = k
. . . . | |   & P((x when e@0),val(e@0)))
. . . . | | by Assert ((r when e') = (r after pred(e')))
. . . . | |
. . . . | |\
. . . . | | .....assertion .....
. . . . | | 1. T : Type
. . . . | | 2. A : Type
. . . . | | 3. P : A→T→𝔹
. . . . | | 4. k : Knd
. . . . | | 5. i : Id
. . . . | | 6. r : Id
. . . . | | 7. x : Id
. . . . | | 8. A
. . . . | | 9. T
```

```
. . . . | | 10. ¬x = r
. . . . | | 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . . | | 12. D' : Dsys
. . . . | | 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . . | | 14. w : World
. . . . | | 15. p : FairFifo
. . . . | | 16. PossibleWorld(D';w)
. . . . | | 17. vartype(i;r) ⊆r 𝔹
. . . . | | 18. ∀e:E. loc(e) = i ⇒ first(e) ⇒ (r when e) = false₂
. . . . | | 19. ∀e:E.
. . . . | | |   loc(e) = i ⇒ ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . . | | 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v) ∨₂ r)
. . . . | | |   ∈ Dsys
. . . . | | 21. vartype(i;r) ⊆r 𝔹
. . . . | | 22. vartype(i;x) ⊆r A
. . . . | | 23. ∀e:E. loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . . | | 24. ∀e:E.
. . . . | | |   loc(e) = i
. . . . | | |   ⇒
. . . . | | |   kind(e) = k
. . . . | | |   ⇒
. . . . | | |   (r after e) = (P((x when e),val(e)) ∨₂ (r when e))
. . . . | | 25. e' : E
. . . . | | 26. loc(e') = i
. . . . | | 27. ¬first(e')
. . . . | | 28. (r after pred(e'))
. . . . | | |   ⟺
. . . . | | |   (∃e@0:E.
. . . . | | |    (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | | |    & kind(e@0) = k
. . . . | | |    & P((x when e@0),val(e@0)))
. . . . | | ⊢  (r when e') = (r after pred(e'))
. . . . | | | by InstESAxiom ES(w) [e';r] 6
. . . . | | |
. . . . | | 1. T : Type
. . . . | | 2. A : Type
. . . . | | 3. P : A→T→𝔹
. . . . | | 4. k : Knd
. . . . | | 5. i : Id
. . . . | | 6. r : Id
. . . . | | 7. x : Id
. . . . | | 8. A
. . . . | | 9. T
. . . . | | 10. ¬x = r
. . . . | | 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . . | | 12. D' : Dsys
. . . . | | 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . . | | 14. w : World
. . . . | | 15. p : FairFifo
. . . . | | 16. PossibleWorld(D';w)
. . . . | | 17. vartype(i;r) ⊆r 𝔹
. . . . | | 18. ∀e:E. loc(e) = i ⇒ first(e) ⇒ (r when e) = false₂
. . . . | | 19. ∀e:E.
. . . . | |     loc(e) = i ⇒ ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . . | | 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v) ∨₂ r)
```

```
. . . . | |     ∈ Dsys
. . . . | | 21. vartype(i;r) ⊆r 𝔹
. . . . | | 22. vartype(i;x) ⊆r A
. . . . | | 23. ∀e:E. loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . . | | 24. ∀e:E.
. . . . | |     loc(e) = i
. . . . | |     ⇒
. . . . | |     kind(e) = k
. . . . | |     ⇒
. . . . | |     (r after e) = (P((x when e),val(e)) ∨2 (r when e))
. . . . | | 25. e' : E
. . . . | | 26. loc(e') = i
. . . . | | 27. ¬first(e')
. . . . | | 28. (r after pred(e'))
. . . . | |     ⟺
. . . . | |     (∃e@0:E.
. . . . | |      (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | |      & kind(e@0) = k
. . . . | |      & P((x when e@0),val(e@0)))
. . . . | | 29. (r when e') = (r after pred(e'))
. . . . | | ⊢  (r when e') = (r after pred(e'))
. . . . | |   by RevHypSubst -1 0 THEN MaAuto THEN UseVartypeHyp
. . . . | |
. . . . | .\
. . . . | . 1. T : Type
. . . . | . 2. A : Type
. . . . | . 3. P : A→T→𝔹
. . . . | . 4. k : Knd
. . . . | . 5. i : Id
. . . . | . 6. r : Id
. . . . | . 7. x : Id
. . . . | . 8. A
. . . . | . 9. T
. . . . | . 10. ¬x = r
. . . . | . 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . . | . 12. D' : Dsys
. . . . | . 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . . | . 14. w : World
. . . . | . 15. p : FairFifo
. . . . | . 16. PossibleWorld(D';w)
. . . . | . 17. vartype(i;r) ⊆r 𝔹
. . . . | . 18. ∀e:E. loc(e) = i ⇒ first(e) ⇒ (r when e) = false2
. . . . | . 19. ∀e:E.
. . . . | . |   loc(e) = i ⇒ ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . . | . 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v) ∨2 r)
. . . . | . |   ∈ Dsys
. . . . | . 21. vartype(i;r) ⊆r 𝔹
. . . . | . 22. vartype(i;x) ⊆r A
. . . . | . 23. ∀e:E. loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . . | . 24. ∀e:E.
. . . . | . |   loc(e) = i
. . . . | . |   ⇒
. . . . | . |   kind(e) = k
. . . . | . |   ⇒
. . . . | . |   (r after e) = (P((x when e),val(e)) ∨2 (r when e))
```

```
. . . . | . 25. e' : E
. . . . | . 26. loc(e') = i
. . . . | . 27. ¬first(e')
. . . . | . 28. (r after pred(e'))
. . . . | . |    ⟺
. . . . | . |    (∃e@0:E.
. . . . | . |     (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . |     & kind(e@0) = k
. . . . | . |     & P((x when e@0),val(e@0)))
. . . . | . 29. (r when e') = (r after pred(e'))
. . . . | . ⊢  (r after e')
. . . . | . |   ⟺
. . . . | . |   (∃e@0:E.
. . . . | . |    (e@0 <loc e') ∨ e@0 = e'
. . . . | . |    & kind(e@0) = k
. . . . | . |    & P((x when e@0),val(e@0)))  by MaAuto
. . . . | . |

. . . . | . |\
. . . . | . | 1. T : Type
. . . . | . | 2. A : Type
. . . . | . | 3. P : A→T→𝔹
. . . . | . | 4. k : Knd
. . . . | . | 5. i : Id
. . . . | . | 6. r : Id
. . . . | . | 7. x : Id
. . . . | . | 8. A
. . . . | . | 9. T
. . . . | . | 10. ¬x = r
. . . . | . | 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . . | . | 12. D' : Dsys
. . . . | . | 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . . | . | 14. w : World
. . . . | . | 15. p : FairFifo
. . . . | . | 16. PossibleWorld(D';w)
. . . . | . | 17. vartype(i;r) ⊆r 𝔹
. . . . | . | 18. ∀e:E. loc(e) = i ⇒ first(e) ⇒ (r when e) = false_2
. . . . | . | 19. ∀e:E.
. . . . | . | |   loc(e) = i
. . . . | . | |   ⇒
. . . . | . | |   ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . . | . | 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v)
. . . . | . | |   ∨_2 r)
. . . . | . | |   ∈ Dsys
. . . . | . | 21. vartype(i;r) ⊆r 𝔹
. . . . | . | 22. vartype(i;x) ⊆r A
. . . . | . | 23. ∀e:E. loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . . | . | 24. ∀e:E.
. . . . | . | |   loc(e) = i
. . . . | . | |   ⇒
. . . . | . | |   kind(e) = k
. . . . | . | |   ⇒
. . . . | . | |   (r after e) = (P((x when e),val(e)) ∨_2 (r when e))
. . . . | . | 25. e' : E
. . . . | . | 26. loc(e') = i
. . . . | . | 27. ¬first(e')
```

```
. . . . | . | 28. (r after pred(e'))
. . . . | . | |    ⇒
. . . . | . | |    (∃e@0:E.
. . . . | . | |     (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . | |     & kind(e@0) = k
. . . . | . | |     & P((x when e@0),val(e@0)))
. . . . | . | 29. (r after pred(e')) ⇐ (∃e@0:E.
. . . . | . | |    (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . | |    & kind(e@0) = k
. . . . | . | |    & P((x when e@0),val(e@0)))
. . . . | . | 30. (r when e') = (r after pred(e'))
. . . . | . | 31. (r after e')
. . . . | . | ⊢  ∃e@0:E.
. . . . | . | |  (e@0 <loc e') ∨ e@0 = e'
. . . . | . | |  & kind(e@0) = k
. . . . | . | |  & P((x when e@0),val(e@0))
. . . . | . | | by Decide (r after pred(e')) THENA MaAuto
. . . . | . | |
. . . . | . | |\
. . . . | . | | 1. T : Type
. . . . | . | | 2. A : Type
. . . . | . | | 3. P : A→T→𝔹
. . . . | . | | 4. k : Knd
. . . . | . | | 5. i : Id
. . . . | . | | 6. r : Id
. . . . | . | | 7. x : Id
. . . . | . | | 8. A
. . . . | . | | 9. T
. . . . | . | | 10. ¬x = r
. . . . | . | | 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . . | . | | 12. D' : Dsys
. . . . | . | | 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . . | . | | 14. w : World
. . . . | . | | 15. p : FairFifo
. . . . | . | | 16. PossibleWorld(D';w)
. . . . | . | | 17. vartype(i;r) ⊆r 𝔹
. . . . | . | | 18. ∀e:E.
. . . . | . | | |   loc(e) = i ⇒ first(e) ⇒ (r when e) = false2
. . . . | . | | 19. ∀e:E.
. . . . | . | | |   loc(e) = i
. . . . | . | | |   ⇒
. . . . | . | | |   ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . . | . | | 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v)
. . . . | . | | |   ∨2 r)
. . . . | . | | |   ∈ Dsys
. . . . | . | | 21. vartype(i;r) ⊆r 𝔹
. . . . | . | | 22. vartype(i;x) ⊆r A
. . . . | . | | 23. ∀e:E.
. . . . | . | | |   loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . . | . | | 24. ∀e:E.
. . . . | . | | |   loc(e) = i
. . . . | . | | |   ⇒
. . . . | . | | |   kind(e) = k
. . . . | . | | |   ⇒
. . . . | . | | |   (r after e) = (P((x when e),val(e)) ∨2 (r when e))
```

```
. . . . | . | | 25. e' : E
. . . . | . | | 26. loc(e') = i
. . . . | . | | 27. ¬first(e')
. . . . | . | | 28. (r after pred(e'))
. . . . | . | | |   ⇒
. . . . | . | | |   (∃e@0:E.
. . . . | . | | |    (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . | | |    & kind(e@0) = k
. . . . | . | | |    & P((x when e@0),val(e@0)))
. . . . | . | | 29. (r after pred(e')) ⇐ (∃e@0:E.
. . . . | . | | |   (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . | | |   & kind(e@0) = k
. . . . | . | | |   & P((x when e@0),val(e@0)))
. . . . | . | | 30. (r when e') = (r after pred(e'))
. . . . | . | | 31. (r after e')
. . . . | . | | 32. (r after pred(e'))
. . . . | . | | ⊢  ∃e@0:E.
. . . . | . | | |  (e@0 <loc e') ∨ e@0 = e'
. . . . | . | | |  & kind(e@0) = k
. . . . | . | | |  & P((x when e@0),val(e@0))  by ThinTrivial
. . . . | . | | |
. . . . | . | | 1. T : Type
. . . . | . | | 2. A : Type
. . . . | . | | 3. P : A→T→𝔹
. . . . | . | | 4. k : Knd
. . . . | . | | 5. i : Id
. . . . | . | | 6. r : Id
. . . . | . | | 7. x : Id
. . . . | . | | 8. A
. . . . | . | | 9. T
. . . . | . | | 10. ¬x = r
. . . . | . | | 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . . | . | | 12. D' : Dsys
. . . . | . | | 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . . | . | | 14. w : World
. . . . | . | | 15. p : FairFifo
. . . . | . | | 16. PossibleWorld(D';w)
. . . . | . | | 17. vartype(i;r) ⊆r 𝔹
. . . . | . | | 18. ∀e:E.
. . . . | . | | |   loc(e) = i ⇒ first(e) ⇒ (r when e) = false2
. . . . | . | | 19. ∀e:E.
. . . . | . | | |   loc(e) = i
. . . . | . | | |   ⇒
. . . . | . | | |   ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . . | . | | 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v)
. . . . | . | | |   ∨2 r)
. . . . | . | | |   ∈ Dsys
. . . . | . | | 21. vartype(i;r) ⊆r 𝔹
. . . . | . | | 22. vartype(i;x) ⊆r A
. . . . | . | | 23. ∀e:E.
. . . . | . | | |   loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . . | . | | 24. ∀e:E.
. . . . | . | | |   loc(e) = i
. . . . | . | | |   ⇒
. . . . | . | | |   kind(e) = k
```

```
. . . . | . | | |   ⇒
. . . . | . | | |   (r after e) = (P((x when e),val(e)) ∨2 (r when e))
. . . . | . | | 25. e' : E
. . . . | . | | 26. loc(e') = i
. . . . | . | | 27. ¬first(e')
. . . . | . | | 28. (r after pred(e')) ⇐ (∃e@0:E.
. . . . | . | | |   (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . | | |   & kind(e@0) = k
. . . . | . | | |   & P((x when e@0),val(e@0)))
. . . . | . | | 29. (r when e') = (r after pred(e'))
. . . . | . | | 30. (r after e')
. . . . | . | | 31. (r after pred(e'))
. . . . | . | | 32. ∃e@0:E.
. . . . | . | | |   (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . | | |   & kind(e@0) = k
. . . . | . | | |   & P((x when e@0),val(e@0))
. . . . | . | | ⊢  ∃e@0:E.
. . . . | . | | |  (e@0 <loc e') ∨ e@0 = e'
. . . . | . | | |  & kind(e@0) = k
. . . . | . | | |  & P((x when e@0),val(e@0))
. . . . | . | | | by ParallelOp -1 THEN ExRepD THEN MaAuto
. . . . | . | | |
. . . . | . | | 1. T : Type
. . . . | . | | 2. A : Type
. . . . | . | | 3. P : A→T→𝔹
. . . . | . | | 4. k : Knd
. . . . | . | | 5. i : Id
. . . . | . | | 6. r : Id
. . . . | . | | 7. x : Id
. . . . | . | | 8. A
. . . . | . | | 9. T
. . . . | . | | 10. ¬x = r
. . . . | . | | 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . . | . | | 12. D' : Dsys
. . . . | . | | 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . . | . | | 14. w : World
. . . . | . | | 15. p : FairFifo
. . . . | . | | 16. PossibleWorld(D';w)
. . . . | . | | 17. vartype(i;r) ⊆r 𝔹
. . . . | . | | 18. ∀e:E.
. . . . | . | | |   loc(e) = i ⇒ first(e) ⇒ (r when e) = false2
. . . . | . | | 19. ∀e:E.
. . . . | . | | |   loc(e) = i
. . . . | . | | |   ⇒
. . . . | . | | |   ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . . | . | | 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v)
. . . . | . | | |   ∨2 r)
. . . . | . | | |   ∈ Dsys
. . . . | . | | 21. vartype(i;r) ⊆r 𝔹
. . . . | . | | 22. vartype(i;x) ⊆r A
. . . . | . | | 23. ∀e:E.
. . . . | . | | |   loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . . | . | | 24. ∀e:E.
. . . . | . | | |   loc(e) = i
. . . . | . | | |   ⇒
```

```
. . . . | . | | |   kind(e) = k
. . . . | . | | |   ⇒
. . . . | . | | |   (r after e) = (P((x when e),val(e)) ∨2 (r when e))
. . . . | . | | | 25. e' : E
. . . . | . | | | 26. loc(e') = i
. . . . | . | | | 27. ¬first(e')
. . . . | . | | | 28. (r after pred(e')) ⇐ (∃e@0:E.
. . . . | . | | |   (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . | | |   & kind(e@0) = k
. . . . | . | | |   & P((x when e@0),val(e@0)))
. . . . | . | | | 29. (r when e') = (r after pred(e'))
. . . . | . | | | 30. (r after e')
. . . . | . | | | 31. (r after pred(e'))
. . . . | . | | | 32. e@0 : E
. . . . | . | | | 33. (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . | | | 34. kind(e@0) = k
. . . . | . | | | 35. P((x when e@0),val(e@0))
. . . . | . | | ⊢  (e@0 <loc e') ∨ e@0 = e'  by OrLeft
. . . . | . | | |
. . . . | . | | | 1. T : Type
. . . . | . | | | 2. A : Type
. . . . | . | | | 3. P : A→T→𝔹
. . . . | . | | | 4. k : Knd
. . . . | . | | | 5. i : Id
. . . . | . | | | 6. r : Id
. . . . | . | | | 7. x : Id
. . . . | . | | | 8. A
. . . . | . | | | 9. T
. . . . | . | | | 10. ¬x = r
. . . . | . | | | 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . . | . | | | 12. D' : Dsys
. . . . | . | | | 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . . | . | | | 14. w : World
. . . . | . | | | 15. p : FairFifo
. . . . | . | | | 16. PossibleWorld(D';w)
. . . . | . | | | 17. vartype(i;r) ⊆r 𝔹
. . . . | . | | | 18. ∀e:E.
. . . . | . | | |   loc(e) = i ⇒ first(e) ⇒ (r when e) = false2
. . . . | . | | | 19. ∀e:E.
. . . . | . | | |   loc(e) = i
. . . . | . | | |   ⇒
. . . . | . | | |   ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . . | . | | | 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v)
. . . . | . | | |   ∨2 r)
. . . . | . | | |   ∈ Dsys
. . . . | . | | | 21. vartype(i;r) ⊆r 𝔹
. . . . | . | | | 22. vartype(i;x) ⊆r A
. . . . | . | | | 23. ∀e:E.
. . . . | . | | |   loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . . | . | | | 24. ∀e:E.
. . . . | . | | |   loc(e) = i
. . . . | . | | |   ⇒
. . . . | . | | |   kind(e) = k
. . . . | . | | |   ⇒
. . . . | . | | |   (r after e) = (P((x when e),val(e)) ∨2 (r when e))
```

```
. . . . | . | | 25. e' : E
. . . . | . | | 26. loc(e') = i
. . . . | . | | 27. ¬first(e')
. . . . | . | | 28. (r after pred(e')) ⇐ (∃e@0:E.
. . . . | . | | |    (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . | | |    & kind(e@0) = k
. . . . | . | | |    & P((x when e@0),val(e@0)))
. . . . | . | | 29. (r when e') = (r after pred(e'))
. . . . | . | | 30. (r after e')
. . . . | . | | 31. (r after pred(e'))
. . . . | . | | 32. e@0 : E
. . . . | . | | 33. (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . | | 34. kind(e@0) = k
. . . . | . | | 35. P((x when e@0),val(e@0))
. . . . | . | | ⊢  (e@0 <loc e')
. . . . | . | | | by RWO
. . . . | . | | |     Thm* ∀the_es:ES, e,e':E.
. . . . | . | | |          (e <loc e')
. . . . | . | | |          ⟺
. . . . | . | | |          ¬first(e')
. . . . | . | | |          & e = pred(e') ∈ E ∨ (e <loc pred(e'))
. . . . | . | | |     {es_DASH_locl_DASH_iff}
. . . . | . | | |     0
. . . . | . | | |
. . . . | . | | 1. T : Type
. . . . | . | | 2. A : Type
. . . . | . | | 3. P : A→T→𝔹
. . . . | . | | 4. k : Knd
. . . . | . | | 5. i : Id
. . . . | . | | 6. r : Id
. . . . | . | | 7. x : Id
. . . . | . | | 8. A
. . . . | . | | 9. T
. . . . | . | | 10. ¬x = r
. . . . | . | | 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . . | . | | 12. D' : Dsys
. . . . | . | | 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . . | . | | 14. w : World
. . . . | . | | 15. p : FairFifo
. . . . | . | | 16. PossibleWorld(D';w)
. . . . | . | | 17. vartype(i;r) ⊆r 𝔹
. . . . | . | | 18. ∀e:E.
. . . . | . | |     loc(e) = i ⇒ first(e) ⇒ (r when e) = false2
. . . . | . | | 19. ∀e:E.
. . . . | . | |     loc(e) = i
. . . . | . | |     ⇒
. . . . | . | |     ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . . | . | | 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v)
. . . . | . | |     ∨2 r)
. . . . | . | |     ∈ Dsys
. . . . | . | | 21. vartype(i;r) ⊆r 𝔹
. . . . | . | | 22. vartype(i;x) ⊆r A
. . . . | . | | 23. ∀e:E.
. . . . | . | |     loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . . | . | | 24. ∀e:E.
```

```
. . . . | . | |     loc(e) = i
. . . . | . | |     ⇒
. . . . | . | |     kind(e) = k
. . . . | . | |     ⇒
. . . . | . | |     (r after e) = (P((x when e),val(e)) ∨₂ (r when e))
. . . . | . | | 25. e' : E
. . . . | . | | 26. loc(e') = i
. . . . | . | | 27. ¬first(e')
. . . . | . | | 28. (r after pred(e')) ⇐ (∃e@0:E.
. . . . | . | |     (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . | |     & kind(e@0) = k
. . . . | . | |     & P((x when e@0),val(e@0)))
. . . . | . | | 29. (r when e') = (r after pred(e'))
. . . . | . | | 30. (r after e')
. . . . | . | | 31. (r after pred(e'))
. . . . | . | | 32. e@0 : E
. . . . | . | | 33. (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . | | 34. kind(e@0) = k
. . . . | . | | 35. P((x when e@0),val(e@0))
. . . . | . | | ⊢  e@0 = pred(e') ∨ (e@0 <loc pred(e'))
. . . . | . | |   by Analyze -3 THENL [OrRight;OrLeft]
. . . . | . | |
. . . . | . | .\
. . . . | . | . 1. T : Type
. . . . | . | . 2. A : Type
. . . . | . | . 3. P : A→T→𝔹
. . . . | . | . 4. k : Knd
. . . . | . | . 5. i : Id
. . . . | . | . 6. r : Id
. . . . | . | . 7. x : Id
. . . . | . | . 8. A
. . . . | . | . 9. T
. . . . | . | . 10. ¬x = r
. . . . | . | . 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . . | . | . 12. D' : Dsys
. . . . | . | . 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . . | . | . 14. w : World
. . . . | . | . 15. p : FairFifo
. . . . | . | . 16. PossibleWorld(D';w)
. . . . | . | . 17. vartype(i;r) ⊆r 𝔹
. . . . | . | . 18. ∀e:E.
. . . . | . | . |   loc(e) = i ⇒ first(e) ⇒ (r when e) = false₂
. . . . | . | . 19. ∀e:E.
. . . . | . | . |   loc(e) = i
. . . . | . | . |   ⇒
. . . . | . | . |   ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . . | . | . 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v)
. . . . | . | . |   ∨₂ r)
. . . . | . | . |   ∈ Dsys
. . . . | . | . 21. vartype(i;r) ⊆r 𝔹
. . . . | . | . 22. vartype(i;x) ⊆r A
. . . . | . | . 23. ∀e:E.
. . . . | . | . |   loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . . | . | . 24. ∀e:E.
. . . . | . | . |   loc(e) = i
```

```
. . . . | . | . |    ⇒
. . . . | . | . |    kind(e) = k
. . . . | . | . |    ⇒
. . . . | . | . |    (r after e) = (P((x when e),val(e)) ∨2 (r when e))
. . . . | . | . 25. e' : E
. . . . | . | . 26. loc(e') = i
. . . . | . | . 27. ¬first(e')
. . . . | . | . 28. (r after pred(e'))
. . . . | . | . |    ⇒
. . . . | . | . |    (∃e@0:E.
. . . . | . | . |     (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . | . |     & kind(e@0) = k
. . . . | . | . |     & P((x when e@0),val(e@0)))
. . . . | . | . 29. (r after pred(e')) ⇐ (∃e@0:E.
. . . . | . | . |    (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . | . |    & kind(e@0) = k
. . . . | . | . |    & P((x when e@0),val(e@0)))
. . . . | . | . 30. (r when e') = (r after pred(e'))
. . . . | . | . 31. (r after e')
. . . . | . | . 32. ¬(r after pred(e'))
. . . . | . | . ⊢  ∃e@0:E.
. . . . | . | . |  (e@0 <loc e') ∨ e@0 = e'
. . . . | . | . |  & kind(e@0) = k
. . . . | . | . |  & P((x when e@0),val(e@0))
. . . . | . | . | by Assert (¬(r after e') = (r when e'))
. . . . | . | . |
. . . . | . | . |\
. . . . | . | . | .....assertion .....
. . . . | . | . | 1. T : Type
. . . . | . | . | 2. A : Type
. . . . | . | . | 3. P : A→T→𝔹
. . . . | . | . | 4. k : Knd
. . . . | . | . | 5. i : Id
. . . . | . | . | 6. r : Id
. . . . | . | . | 7. x : Id
. . . . | . | . | 8. A
. . . . | . | . | 9. T
. . . . | . | . | 10. ¬x = r
. . . . | . | . | 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . . | . | . | 12. D' : Dsys
. . . . | . | . | 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . . | . | . | 14. w : World
. . . . | . | . | 15. p : FairFifo
. . . . | . | . | 16. PossibleWorld(D';w)
. . . . | . | . | 17. vartype(i;r) ⊆r 𝔹
. . . . | . | . | 18. ∀e:E.
. . . . | . | . |     loc(e) = i ⇒ first(e) ⇒ (r when e) = false2
. . . . | . | . | 19. ∀e:E.
. . . . | . | . |     loc(e) = i
. . . . | . | . |     ⇒
. . . . | . | . |     ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . . | . | . | 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v)
. . . . | . | . |     ∨2 r)
. . . . | . | . |     ∈ Dsys
. . . . | . | . | 21. vartype(i;r) ⊆r 𝔹
```

```
. . . . | . | . | 22. vartype(i;x) ⊆r A
. . . . | . | . | 23. ∀e:E.
. . . . | . | . |     loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . . | . | . | 24. ∀e:E.
. . . . | . | . |     loc(e) = i
. . . . | . | . |     ⇒
. . . . | . | . |     kind(e) = k
. . . . | . | . |     ⇒
. . . . | . | . |     (r after e)
. . . . | . | . |     =
. . . . | . | . |     (P((x when e),val(e)) ∨2 (r when e))
. . . . | . | . | 25. e' : E
. . . . | . | . | 26. loc(e') = i
. . . . | . | . | 27. ¬first(e')
. . . . | . | . | 28. (r after pred(e'))
. . . . | . | . |     ⇒
. . . . | . | . |     (∃e@0:E.
. . . . | . | . |      (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . | . |      & kind(e@0) = k
. . . . | . | . |      & P((x when e@0),val(e@0)))
. . . . | . | . | 29. (r after pred(e')) ⇐ (∃e@0:E.
. . . . | . | . |     (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . | . |     & kind(e@0) = k
. . . . | . | . |     & P((x when e@0),val(e@0)))
. . . . | . | . | 30. (r when e') = (r after pred(e'))
. . . . | . | . | 31. (r after e')
. . . . | . | . | 32. ¬(r after pred(e'))
. . . . | . | . | ⊢  ¬(r after e') = (r when e')
. . . . | . | . |   by MaBoolCase (r after e')
. . . . | . | . |      THEN
. . . . | . | . |      MaBoolCase (r when e')
. . . . | . | . |      THEN
. . . . | . | . |      MaBoolCase (r after pred(e'))
. . . . | . | . |
. . . . | . | . .\
. . . . | . | . . 1. T : Type
. . . . | . | . . 2. A : Type
. . . . | . | . . 3. P : A→T→𝔹
. . . . | . | . . 4. k : Knd
. . . . | . | . . 5. i : Id
. . . . | . | . . 6. r : Id
. . . . | . | . . 7. x : Id
. . . . | . | . . 8. A
. . . . | . | . . 9. T
. . . . | . | . . 10. ¬x = r
. . . . | . | . . 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . . | . | . . 12. D' : Dsys
. . . . | . | . . 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . . | . | . . 14. w : World
. . . . | . | . . 15. p : FairFifo
. . . . | . | . . 16. PossibleWorld(D';w)
. . . . | . | . . 17. vartype(i;r) ⊆r 𝔹
. . . . | . | . . 18. ∀e:E.
. . . . | . | . . |   loc(e) = i ⇒ first(e) ⇒ (r when e) = false2
. . . . | . | . . 19. ∀e:E.
```

```
. . . . | . | . . . |   loc(e) = i
. . . . | . | . . . |   ⇒
. . . . | . | . . . |   ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . . | . | . . . 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v)
. . . . | . | . . . |   ∨₂ r)
. . . . | . | . . . |   ∈ Dsys
. . . . | . | . . . 21. vartype(i;r) ⊆r 𝔹
. . . . | . | . . . 22. vartype(i;x) ⊆r A
. . . . | . | . . . 23. ∀e:E.
. . . . | . | . . . |   loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . . | . | . . . 24. ∀e:E.
. . . . | . | . . . |   loc(e) = i
. . . . | . | . . . |   ⇒
. . . . | . | . . . |   kind(e) = k
. . . . | . | . . . |   ⇒
. . . . | . | . . . |   (r after e)
. . . . | . | . . . |   =
. . . . | . | . . . |   (P((x when e),val(e)) ∨₂ (r when e))
. . . . | . | . . . 25. e' : E
. . . . | . | . . . 26. loc(e') = i
. . . . | . | . . . 27. ¬first(e')
. . . . | . | . . . 28. (r after pred(e'))
. . . . | . | . . . |   ⇒
. . . . | . | . . . |   (∃e@0:E.
. . . . | . | . . . |    (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . | . . . |    & kind(e@0) = k
. . . . | . | . . . |    & P((x when e@0),val(e@0)))
. . . . | . | . . . 29. (r after pred(e')) ⇐ (∃e@0:E.
. . . . | . | . . . |   (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . | . . . |   & kind(e@0) = k
. . . . | . | . . . |   & P((x when e@0),val(e@0)))
. . . . | . | . . . 30. (r when e') = (r after pred(e'))
. . . . | . | . . . 31. (r after e')
. . . . | . | . . . 32. ¬(r after pred(e'))
. . . . | . | . . . 33. ¬(r after e') = (r when e')
. . . . | . | . . . ⊢  ∃e@0:E.
. . . . | . | . . . |  (e@0 <loc e') ∨ e@0 = e'
. . . . | . | . . . |  & kind(e@0) = k
. . . . | . | . . . |  & P((x when e@0),val(e@0))
. . . . | . | . . . | by RepeatFor 2
. . . . | . | . . . |    (AllHyps
. . . . | . | . . . |     (λh.InstHyp [e'] h THENA Complete Auto))
. . . . | . | . . . |
. . . . | . | . . . 1. T : Type
. . . . | . | . . . 2. A : Type
. . . . | . | . . . 3. P : A→T→𝔹
. . . . | . | . . . 4. k : Knd
. . . . | . | . . . 5. i : Id
. . . . | . | . . . 6. r : Id
. . . . | . | . . . 7. x : Id
. . . . | . | . . . 8. A
. . . . | . | . . . 9. T
. . . . | . | . . . 10. ¬x = r
. . . . | . | . . . 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . . | . | . . . 12. D' : Dsys
```

```
. . . . | . | . . 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . . | . | . . 14. w : World
. . . . | . | . . 15. p : FairFifo
. . . . | . | . . 16. PossibleWorld(D';w)
. . . . | . | . . 17. vartype(i;r) ⊆r 𝔹
. . . . | . | . . 18. ∀e:E.
. . . . | . | . . |   loc(e) = i ⇒ first(e) ⇒ (r when e) = false₂
. . . . | . | . . 19. ∀e:E.
. . . . | . | . . |   loc(e) = i
. . . . | . | . . |   ⇒
. . . . | . | . . |   ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . . | . | . . 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v)
. . . . | . | . . |   ∨₂ r)
. . . . | . | . . |   ∈ Dsys
. . . . | . | . . 21. vartype(i;r) ⊆r 𝔹
. . . . | . | . . 22. vartype(i;x) ⊆r A
. . . . | . | . . 23. ∀e:E.
. . . . | . | . . |   loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . . | . | . . 24. ∀e:E.
. . . . | . | . . |   loc(e) = i
. . . . | . | . . |   ⇒
. . . . | . | . . |   kind(e) = k
. . . . | . | . . |   ⇒
. . . . | . | . . |   (r after e)
. . . . | . | . . |   =
. . . . | . | . . |   (P((x when e),val(e)) ∨₂ (r when e))
. . . . | . | . . 25. e' : E
. . . . | . | . . 26. loc(e') = i
. . . . | . | . . 27. ¬first(e')
. . . . | . | . . 28. (r after pred(e'))
. . . . | . | . . |   ⇒
. . . . | . | . . |   (∃e@0:E.
. . . . | . | . . |    (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . | . . |    & kind(e@0) = k
. . . . | . | . . |    & P((x when e@0),val(e@0)))
. . . . | . | . . 29. (r after pred(e')) ⇐ (∃e@0:E.
. . . . | . | . . |   (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . | . . |   & kind(e@0) = k
. . . . | . | . . |   & P((x when e@0),val(e@0)))
. . . . | . | . . 30. (r when e') = (r after pred(e'))
. . . . | . | . . 31. (r after e')
. . . . | . | . . 32. ¬(r after pred(e'))
. . . . | . | . . 33. ¬(r after e') = (r when e')
. . . . | . | . . 34. kind(e') = k
. . . . | . | . . 35. (r after e')
. . . . | . | . . |   =
. . . . | . | . . |   (P((x when e'),val(e')) ∨₂ (r when e'))
. . . . | . | . . 36. valtype(e') ⊆r T
. . . . | . | . . 37. kind(e') = k
. . . . | . | . . ⊢  ∃e@0:E.
. . . . | . | . . |  (e@0 <loc e') ∨ e@0 = e'
. . . . | . | . . |  & kind(e@0) = k
. . . . | . | . . |  & P((x when e@0),val(e@0))
. . . . | . | . . | by InstConcl [e'] THEN MaAuto
. . . . | . | . . |
```

```
. . . . | . | . . |\
. . . . | . | . . | 1. T : Type
. . . . | . | . . | 2. A : Type
. . . . | . | . . | 3. P : A→T→𝔹
. . . . | . | . . | 4. k : Knd
. . . . | . | . . | 5. i : Id
. . . . | . | . . | 6. r : Id
. . . . | . | . . | 7. x : Id
. . . . | . | . . | 8. A
. . . . | . | . . | 9. T
. . . . | . | . . | 10. ¬x = r
. . . . | . | . . | 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)))
. . . . | . | . . |     ∈ Dsys
. . . . | . | . . | 12. D' : Dsys
. . . . | . | . . | 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . . | . | . . | 14. w : World
. . . . | . | . . | 15. p : FairFifo
. . . . | . | . . | 16. PossibleWorld(D';w)
. . . . | . | . . | 17. vartype(i;r) ⊆r 𝔹
. . . . | . | . . | 18. ∀e:E.
. . . . | . | . . |     loc(e) = i ⇒ first(e) ⇒ (r when e) = false_2
. . . . | . | . . | 19. ∀e:E.
. . . . | . | . . |     loc(e) = i
. . . . | . | . . |     ⇒
. . . . | . | . . |     ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . . | . | . . | 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P
. . . . | . | . . |                                                (x
. . . . | . | . . |                                                ,v)
. . . . | . | . . |     ∨_2 r)
. . . . | . | . . |     ∈ Dsys
. . . . | . | . . | 21. vartype(i;r) ⊆r 𝔹
. . . . | . | . . | 22. vartype(i;x) ⊆r A
. . . . | . | . . | 23. ∀e:E.
. . . . | . | . . |     loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . . | . | . . | 24. ∀e:E.
. . . . | . | . . |     loc(e) = i
. . . . | . | . . |     ⇒
. . . . | . | . . |     kind(e) = k
. . . . | . | . . |     ⇒
. . . . | . | . . |     (r after e)
. . . . | . | . . |     =
. . . . | . | . . |     (P((x when e),val(e)) ∨_2 (r when e))
. . . . | . | . . | 25. e' : E
. . . . | . | . . | 26. loc(e') = i
. . . . | . | . . | 27. ¬first(e')
. . . . | . | . . | 28. (r after pred(e'))
. . . . | . | . . |     ⇒
. . . . | . | . . |     (∃e@0:E.
. . . . | . | . . |      (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . | . . |      & kind(e@0) = k
. . . . | . | . . |      & P((x when e@0),val(e@0)))
. . . . | . | . . | 29. (r after pred(e')) ⇐ (∃e@0:E.
. . . . | . | . . |     (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . | . . |     & kind(e@0) = k
. . . . | . | . . |     & P((x when e@0),val(e@0)))
```

```
. . . . | . | . . . | 30. (r when e') = (r after pred(e'))
. . . . | . | . . . | 31. (r after e')
. . . . | . | . . . | 32. ¬(r after pred(e'))
. . . . | . | . . . | 33. ¬(r after e') = (r when e')
. . . . | . | . . . | 34. kind(e') = k
. . . . | . | . . . | 35. (r after e')
. . . . | . | . . . |     =
. . . . | . | . . . |     (P((x when e'),val(e')) ∨2 (r when e'))
. . . . | . | . . . | 36. valtype(e') ⊆r T
. . . . | . | . . . | 37. kind(e') = k
. . . . | . | . . . | ⊢  (e' <loc e') ∨ e' = e'  by OrRight
. . . . | . | . . . |
. . . . | . | . . . .\
. . . . | . | . . . . 1. T : Type
. . . . | . | . . . . 2. A : Type
. . . . | . | . . . . 3. P : A→T→𝔹
. . . . | . | . . . . 4. k : Knd
. . . . | . | . . . . 5. i : Id
. . . . | . | . . . . 6. r : Id
. . . . | . | . . . . 7. x : Id
. . . . | . | . . . . 8. A
. . . . | . | . . . . 9. T
. . . . | . | . . . . 10. ¬x = r
. . . . | . | . . . . 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)))
. . . . | . | . . . .     ∈ Dsys
. . . . | . | . . . . 12. D' : Dsys
. . . . | . | . . . . 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . . | . | . . . . 14. w : World
. . . . | . | . . . . 15. p : FairFifo
. . . . | . | . . . . 16. PossibleWorld(D';w)
. . . . | . | . . . . 17. vartype(i;r) ⊆r 𝔹
. . . . | . | . . . . 18. ∀e:E.
. . . . | . | . . . .     loc(e) = i ⇒ first(e) ⇒ (r when e) = false2
. . . . | . | . . . . 19. ∀e:E.
. . . . | . | . . . .     loc(e) = i
. . . . | . | . . . .     ⇒
. . . . | . | . . . .     ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . . | . | . . . . 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P
. . . . | . | . . . .                                              (x
. . . . | . | . . . .                                              ,v)
. . . . | . | . . . .     ∨2 r)
. . . . | . | . . . .     ∈ Dsys
. . . . | . | . . . . 21. vartype(i;r) ⊆r 𝔹
. . . . | . | . . . . 22. vartype(i;x) ⊆r A
. . . . | . | . . . . 23. ∀e:E.
. . . . | . | . . . .     loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . . | . | . . . . 24. ∀e:E.
. . . . | . | . . . .     loc(e) = i
. . . . | . | . . . .     ⇒
. . . . | . | . . . .     kind(e) = k
. . . . | . | . . . .     ⇒
. . . . | . | . . . .     (r after e)
. . . . | . | . . . .     =
. . . . | . | . . . .     (P((x when e),val(e)) ∨2 (r when e))
. . . . | . | . . . . 25. e' : E
```

```
. . . . | . | . . . 26. loc(e’) = i
. . . . | . | . . . 27. ¬first(e’)
. . . . | . | . . . 28. (r after pred(e’))
. . . . | . | . . .     ⇒
. . . . | . | . . .     (∃e@0:E.
. . . . | . | . . .      (e@0 <loc pred(e’)) ∨ e@0 = pred(e’)
. . . . | . | . . .      & kind(e@0) = k
. . . . | . | . . .      & P((x when e@0),val(e@0)))
. . . . | . | . . . 29. (r after pred(e’)) ⇐ (∃e@0:E.
. . . . | . | . . .     (e@0 <loc pred(e’)) ∨ e@0 = pred(e’)
. . . . | . | . . .     & kind(e@0) = k
. . . . | . | . . .     & P((x when e@0),val(e@0)))
. . . . | . | . . . 30. (r when e’) = (r after pred(e’))
. . . . | . | . . . 31. (r after e’)
. . . . | . | . . . 32. ¬(r after pred(e’))
. . . . | . | . . . 33. ¬(r after e’) = (r when e’)
. . . . | . | . . . 34. kind(e’) = k
. . . . | . | . . . 35. (r after e’)
. . . . | . | . . .     =
. . . . | . | . . .     (P((x when e’),val(e’)) ∨₂ (r when e’))
. . . . | . | . . . 36. valtype(e’) ⊆r T
. . . . | . | . . . 37. kind(e’) = k
. . . . | . | . . . ⊢  P((x when e’),val(e’))
. . . . | . | . . .   by MaBoolCase (P((x when e’),val(e’)))
. . . . | . |
. . . . | . .\
. . . . | . . 1. T : Type
. . . . | . . 2. A : Type
. . . . | . . 3. P : A→T→𝔹
. . . . | . . 4. k : Knd
. . . . | . . 5. i : Id
. . . . | . . 6. r : Id
. . . . | . . 7. x : Id
. . . . | . . 8. A
. . . . | . . 9. T
. . . . | . . 10. ¬x = r
. . . . | . . 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . . | . . 12. D’ : Dsys
. . . . | . . 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D’
. . . . | . . 14. w : World
. . . . | . . 15. p : FairFifo
. . . . | . . 16. PossibleWorld(D’;w)
. . . . | . . 17. vartype(i;r) ⊆r 𝔹
. . . . | . . 18. ∀e:E. loc(e) = i ⇒ first(e) ⇒ (r when e) = false₂
. . . . | . . 19. ∀e:E.
. . . . | . . |   loc(e) = i
. . . . | . . |   ⇒
. . . . | . . |   ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . . | . . 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v)
. . . . | . . |   ∨₂ r)
. . . . | . . |   ∈ Dsys
. . . . | . . 21. vartype(i;r) ⊆r 𝔹
. . . . | . . 22. vartype(i;x) ⊆r A
. . . . | . . 23. ∀e:E. loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . . | . . 24. ∀e:E.
```

```
. . . . | . . |   loc(e) = i
. . . . | . . |   ⇒
. . . . | . . |   kind(e) = k
. . . . | . . |   ⇒
. . . . | . . |   (r after e) = (P((x when e),val(e)) ∨2 (r when e))
. . . . | . . 25. e' : E
. . . . | . . 26. loc(e') = i
. . . . | . . 27. ¬first(e')
. . . . | . . 28. (r after pred(e'))
. . . . | . . |   ⇒
. . . . | . . |   (∃e@0:E.
. . . . | . . |    (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . . |    & kind(e@0) = k
. . . . | . . |    & P((x when e@0),val(e@0)))
. . . . | . . 29. (r after pred(e')) ⇐ (∃e@0:E.
. . . . | . . |   (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . . |   & kind(e@0) = k
. . . . | . . |   & P((x when e@0),val(e@0)))
. . . . | . . 30. (r when e') = (r after pred(e'))
. . . . | . . 31. ∃e@0:E.
. . . . | . . |   (e@0 <loc e') ∨ e@0 = e'
. . . . | . . |   & kind(e@0) = k
. . . . | . . |   & P((x when e@0),val(e@0))
. . . . | . . ⊢  (r after e')  by ExRepD
. . . . | . . |
. . . . | . . 1. T : Type
. . . . | . . 2. A : Type
. . . . | . . 3. P : A→T→𝔹
. . . . | . . 4. k : Knd
. . . . | . . 5. i : Id
. . . . | . . 6. r : Id
. . . . | . . 7. x : Id
. . . . | . . 8. A
. . . . | . . 9. T
. . . . | . . 10. ¬x = r
. . . . | . . 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . . | . . 12. D' : Dsys
. . . . | . . 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . . | . . 14. w : World
. . . . | . . 15. p : FairFifo
. . . . | . . 16. PossibleWorld(D';w)
. . . . | . . 17. vartype(i;r) ⊆r 𝔹
. . . . | . . 18. ∀e:E. loc(e) = i ⇒ first(e) ⇒ (r when e) = false2
. . . . | . . 19. ∀e:E.
. . . . | . . |   loc(e) = i
. . . . | . . |   ⇒
. . . . | . . |   ¬(r after e) = (r when e) ⇒ kind(e) = k

. . . . | . . 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v)
. . . . | . . |   ∨2 r)
. . . . | . . |   ∈ Dsys
. . . . | . . 21. vartype(i;r) ⊆r 𝔹
. . . . | . . 22. vartype(i;x) ⊆r A
. . . . | . . 23. ∀e:E. loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . . | . . 24. ∀e:E.
. . . . | . . |   loc(e) = i
```

```
. . . . | . . |   ⇒
. . . . | . . |   kind(e) = k
. . . . | . . |   ⇒
. . . . | . . |   (r after e) = (P((x when e),val(e)) ∨2 (r when e))
. . . . | . . 25. e' : E
. . . . | . . 26. loc(e') = i
. . . . | . . 27. ¬first(e')
. . . . | . . 28. (r after pred(e'))
. . . . | . . |   ⇒
. . . . | . . |   (∃e@0:E.
. . . . | . . |    (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . . |    & kind(e@0) = k
. . . . | . . |    & P((x when e@0),val(e@0)))
. . . . | . . 29. (r after pred(e')) ⇐ (∃e@0:E.
. . . . | . . |   (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . . |   & kind(e@0) = k
. . . . | . . |   & P((x when e@0),val(e@0)))
. . . . | . . 30. (r when e') = (r after pred(e'))
. . . . | . . 31. e@0 : E
. . . . | . . 32. (e@0 <loc e') ∨ e@0 = e'
. . . . | . . 33. kind(e@0) = k
. . . . | . . 34. P((x when e@0),val(e@0))
. . . . | . . ⊢  (r after e')  by Analyze -3
. . . . | . . |
. . . . | . . |\
. . . . | . . | 1. T : Type
. . . . | . . | 2. A : Type
. . . . | . . | 3. P : A→T→𝔹
. . . . | . . | 4. k : Knd
. . . . | . . | 5. i : Id
. . . . | . . | 6. r : Id
. . . . | . . | 7. x : Id
. . . . | . . | 8. A
. . . . | . . | 9. T
. . . . | . . | 10. ¬x = r
. . . . | . . | 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . . | . . | 12. D' : Dsys
. . . . | . . | 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . . | . . | 14. w : World
. . . . | . . | 15. p : FairFifo
. . . . | . . | 16. PossibleWorld(D';w)
. . . . | . . | 17. vartype(i;r) ⊆r 𝔹
. . . . | . . | 18. ∀e:E.
. . . . | . . | |   loc(e) = i ⇒ first(e) ⇒ (r when e) = false2
. . . . | . . | 19. ∀e:E.
. . . . | . . | |   loc(e) = i
. . . . | . . | |   ⇒
. . . . | . . | |   ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . . | . . | 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v)
. . . . | . . | |   ∨2 r)
. . . . | . . | |   ∈ Dsys
. . . . | . . | 21. vartype(i;r) ⊆r 𝔹
. . . . | . . | 22. vartype(i;x) ⊆r A
. . . . | . . | 23. ∀e:E.
. . . . | . . | |   loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
```

```
. . . . | . . | 24. ∀e:E.
. . . . | . . | |   loc(e) = i
. . . . | . . | |   ⇒
. . . . | . . | |   kind(e) = k
. . . . | . . | |   ⇒
. . . . | . . | |   (r after e) = (P((x when e),val(e)) ∨₂ (r when e))
. . . . | . . | 25. e' : E
. . . . | . . | 26. loc(e') = i
. . . . | . . | 27. ¬first(e')
. . . . | . . | 28. (r after pred(e'))
. . . . | . . | |   ⇒
. . . . | . . | |   (∃e@0:E.
. . . . | . . | |    (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . . | |    & kind(e@0) = k
. . . . | . . | |    & P((x when e@0),val(e@0)))
. . . . | . . | 29. (r after pred(e')) ⇐ (∃e@0:E.
. . . . | . . | |   (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . . | |   & kind(e@0) = k
. . . . | . . | |   & P((x when e@0),val(e@0)))
. . . . | . . | 30. (r when e') = (r after pred(e'))
. . . . | . . | 31. e@0 : E
. . . . | . . | 32. (e@0 <loc e')
. . . . | . . | 33. kind(e@0) = k
. . . . | . . | 34. P((x when e@0),val(e@0))
. . . . | . . | ⊢  (r after e')  by Analyze -6
. . . . | . . | |
. . . . | . . | |\
. . . . | . . | | .....antecedent .....
. . . . | . . | | 1. T : Type
. . . . | . . | | 2. A : Type
. . . . | . . | | 3. P : A→T→𝔹
. . . . | . . | | 4. k : Knd
. . . . | . . | | 5. i : Id
. . . . | . . | | 6. r : Id
. . . . | . . | | 7. x : Id
. . . . | . . | | 8. A
. . . . | . . | | 9. T
. . . . | . . | | 10. ¬x = r
. . . . | . . | | 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . . | . . | | 12. D' : Dsys
. . . . | . . | | 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . . | . . | | 14. w : World
. . . . | . . | | 15. p : FairFifo
. . . . | . . | | 16. PossibleWorld(D';w)
. . . . | . . | | 17. vartype(i;r) ⊆r 𝔹
. . . . | . . | | 18. ∀e:E.
. . . . | . . | | |   loc(e) = i ⇒ first(e) ⇒ (r when e) = false₂
. . . . | . . | | 19. ∀e:E.
. . . . | . . | | |   loc(e) = i
. . . . | . . | | |   ⇒
. . . . | . . | | |   ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . . | . . | | 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v)
. . . . | . . | | |   ∨₂ r)
. . . . | . . | | |   ∈ Dsys
. . . . | . . | | 21. vartype(i;r) ⊆r 𝔹
```

```
. . . . | . . | | 22. vartype(i;x) ⊆r A
. . . . | . . | | 23. ∀e:E.
. . . . | . . | | |   loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . . | . . | | 24. ∀e:E.
. . . . | . . | | |   loc(e) = i
. . . . | . . | | |   ⇒
. . . . | . . | | |   kind(e) = k
. . . . | . . | | |   ⇒
. . . . | . . | | |   (r after e)
. . . . | . . | | |   =
. . . . | . . | | |   (P((x when e),val(e)) ∨2 (r when e))
. . . . | . . | | 25. e' : E
. . . . | . . | | 26. loc(e') = i
. . . . | . . | | 27. ¬first(e')
. . . . | . . | | 28. (r after pred(e'))
. . . . | . . | | |   ⇒
. . . . | . . | | |   (∃e@0:E.
. . . . | . . | | |    (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . . | | |    & kind(e@0) = k
. . . . | . . | | |    & P((x when e@0),val(e@0)))
. . . . | . . | | 29. (r when e') = (r after pred(e'))
. . . . | . . | | 30. e@0 : E
. . . . | . . | | 31. (e@0 <loc e')
. . . . | . . | | 32. kind(e@0) = k
. . . . | . . | | 33. P((x when e@0),val(e@0))
. . . . | . . | | ⊢  ∃e@0:E.
. . . . | . . | | |  (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . . | | |  & kind(e@0) = k
. . . . | . . | | |  & P((x when e@0),val(e@0))
. . . . | . . | | | by InstConcl [e@0] THEN MaAuto
. . . . | . . | | |
. . . . | . . | | 1. T : Type
. . . . | . . | | 2. A : Type
. . . . | . . | | 3. P : A→T→𝔹
. . . . | . . | | 4. k : Knd
. . . . | . . | | 5. i : Id
. . . . | . . | | 6. r : Id
. . . . | . . | | 7. x : Id
. . . . | . . | | 8. A
. . . . | . . | | 9. T
. . . . | . . | | 10. ¬x = r
. . . . | . . | | 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . . | . . | | 12. D' : Dsys
. . . . | . . | | 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . . | . . | | 14. w : World
. . . . | . . | | 15. p : FairFifo
. . . . | . . | | 16. PossibleWorld(D';w)
. . . . | . . | | 17. vartype(i;r) ⊆r 𝔹
. . . . | . . | | 18. ∀e:E.
. . . . | . . | | |   loc(e) = i ⇒ first(e) ⇒ (r when e) = false2
. . . . | . . | | 19. ∀e:E.
. . . . | . . | | |   loc(e) = i
. . . . | . . | | |   ⇒
. . . . | . . | | |   ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . . | . . | | 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v)
```

```
. . . . | . . | | |   ∨₂ r)
. . . . | . . | | |   ∈ Dsys
. . . . | . . | | 21. vartype(i;r) ⊆r 𝔹
. . . . | . . | | 22. vartype(i;x) ⊆r A
. . . . | . . | | 23. ∀e:E.
. . . . | . . | | |   loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . . | . . | | 24. ∀e:E.
. . . . | . . | | |   loc(e) = i
. . . . | . . | | |   ⇒
. . . . | . . | | |   kind(e) = k
. . . . | . . | | |   ⇒
. . . . | . . | | |   (r after e)
. . . . | . . | | |   =
. . . . | . . | | |   (P((x when e),val(e)) ∨₂ (r when e))
. . . . | . . | | 25. e' : E
. . . . | . . | | 26. loc(e') = i
. . . . | . . | | 27. ¬first(e')
. . . . | . . | | 28. (r after pred(e'))
. . . . | . . | | |   ⇒
. . . . | . . | | |   (∃e@0:E.
. . . . | . . | | |    (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . . | | |    & kind(e@0) = k
. . . . | . . | | |    & P((x when e@0),val(e@0)))
. . . . | . . | | 29. (r when e') = (r after pred(e'))
. . . . | . . | | 30. e@0 : E
. . . . | . . | | 31. (e@0 <loc e')
. . . . | . . | | 32. kind(e@0) = k
. . . . | . . | | 33. P((x when e@0),val(e@0))
. . . . | . . | | ⊢  (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . . | | | by RW0
. . . . | . . | | |    Thm* ∀the_es:ES, e,e':E.
. . . . | . . | | |         (e <loc e')
. . . . | . . | | |         ⟺
. . . . | . . | | |         ¬first(e')
. . . . | . . | | |         & e = pred(e') ∈ E ∨ (e <loc pred(e'))
. . . . | . . | | |    {es_DASH_locl_DASH_iff}
. . . . | . . | | |    -3
. . . . | . . | | |
. . . . | . . | | 1. T : Type
. . . . | . . | | 2. A : Type
. . . . | . . | | 3. P : A→T→𝔹
. . . . | . . | | 4. k : Knd
. . . . | . . | | 5. i : Id
. . . . | . . | | 6. r : Id
. . . . | . . | | 7. x : Id
. . . . | . . | | 8. A
. . . . | . . | | 9. T
. . . . | . . | | 10. ¬x = r
. . . . | . . | | 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . . | . . | | 12. D' : Dsys
. . . . | . . | | 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . . | . . | | 14. w : World
. . . . | . . | | 15. p : FairFifo
. . . . | . . | | 16. PossibleWorld(D';w)
. . . . | . . | | 17. vartype(i;r) ⊆r 𝔹
```

```
. . . . | . . | | 18. ∀e:E.
. . . . | . . | |     loc(e) = i ⇒ first(e) ⇒ (r when e) = false₂
. . . . | . . | | 19. ∀e:E.
. . . . | . . | |     loc(e) = i
. . . . | . . | |     ⇒
. . . . | . . | |     ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . . | . . | | 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v)
. . . . | . . | |     ∨₂ r)
. . . . | . . | |     ∈ Dsys
. . . . | . . | | 21. vartype(i;r) ⊆r 𝔹
. . . . | . . | | 22. vartype(i;x) ⊆r A
. . . . | . . | | 23. ∀e:E.
. . . . | . . | |     loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . . | . . | | 24. ∀e:E.
. . . . | . . | |     loc(e) = i
. . . . | . . | |     ⇒
. . . . | . . | |     kind(e) = k
. . . . | . . | |     ⇒
. . . . | . . | |     (r after e)
. . . . | . . | |     =
. . . . | . . | |     (P((x when e),val(e)) ∨₂ (r when e))
. . . . | . . | | 25. e' : E
. . . . | . . | | 26. loc(e') = i
. . . . | . . | | 27. ¬first(e')
. . . . | . . | | 28. (r after pred(e'))
. . . . | . . | |     ⇒
. . . . | . . | |     (∃e@0:E.
. . . . | . . | |      (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . . | |      & kind(e@0) = k
. . . . | . . | |      & P((x when e@0),val(e@0)))
. . . . | . . | | 29. (r when e') = (r after pred(e'))
. . . . | . . | | 30. e@0 : E
. . . . | . . | | 31. ¬first(e')
. . . . | . . | | 32. e@0 = pred(e') ∨ (e@0 <loc pred(e'))
. . . . | . . | | 33. kind(e@0) = k
. . . . | . . | | 34. P((x when e@0),val(e@0))
. . . . | . . | | ⊢  (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . . | |   by Analyze -3 THENL [OrRight;OrLeft]
. . . . | . . | |
. . . . | . . | .\
. . . . | . . | . 1. T : Type
. . . . | . . | . 2. A : Type
. . . . | . . | . 3. P : A→T→𝔹
. . . . | . . | . 4. k : Knd
. . . . | . . | . 5. i : Id
. . . . | . . | . 6. r : Id
. . . . | . . | . 7. x : Id
. . . . | . . | . 8. A
. . . . | . . | . 9. T
. . . . | . . | . 10. ¬x = r
. . . . | . . | . 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . . | . . | . 12. D' : Dsys
. . . . | . . | . 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . . | . . | . 14. w : World
. . . . | . . | . 15. p : FairFifo
```

```
. . . . | . . | . 16. PossibleWorld(D';w)
. . . . | . . | . 17. vartype(i;r) ⊆r 𝔹
. . . . | . . | . 18. ∀e:E.
. . . . | . . | . |   loc(e) = i ⇒ first(e) ⇒ (r when e) = false2
. . . . | . . | . 19. ∀e:E.
. . . . | . . | . |   loc(e) = i
. . . . | . . | . |   ⇒
. . . . | . . | . |   ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . . | . . | . 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v)
. . . . | . . | . |   ∨2 r)
. . . . | . . | . |   ∈ Dsys
. . . . | . . | . 21. vartype(i;r) ⊆r 𝔹
. . . . | . . | . 22. vartype(i;x) ⊆r A
. . . . | . . | . 23. ∀e:E.
. . . . | . . | . |   loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . . | . . | . 24. ∀e:E.
. . . . | . . | . |   loc(e) = i
. . . . | . . | . |   ⇒
. . . . | . . | . |   kind(e) = k
. . . . | . . | . |   ⇒
. . . . | . . | . |   (r after e)
. . . . | . . | . |   =
. . . . | . . | . |   (P((x when e),val(e)) ∨2 (r when e))
. . . . | . . | . 25. e' : E
. . . . | . . | . 26. loc(e') = i
. . . . | . . | . 27. ¬first(e')
. . . . | . . | . 28. (r after pred(e'))
. . . . | . . | . |   ⇒
. . . . | . . | . |   (∃e@0:E.
. . . . | . . | . |    (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . . | . |    & kind(e@0) = k
. . . . | . . | . |    & P((x when e@0),val(e@0)))
. . . . | . . | . 29. (r when e') = (r after pred(e'))
. . . . | . . | . 30. e@0 : E
. . . . | . . | . 31. (e@0 <loc e')
. . . . | . . | . 32. kind(e@0) = k
. . . . | . . | . 33. P((x when e@0),val(e@0))
. . . . | . . | . 34. (r after pred(e'))
. . . . | . . | . ⊢  (r after e')  by SupposeNot THEN MaAuto
. . . . | . . | . |
. . . . | . . | . 1. T : Type
. . . . | . . | . 2. A : Type
. . . . | . . | . 3. P : A→T→𝔹
. . . . | . . | . 4. k : Knd
. . . . | . . | . 5. i : Id
. . . . | . . | . 6. r : Id
. . . . | . . | . 7. x : Id
. . . . | . . | . 8. A
. . . . | . . | . 9. T
. . . . | . . | . 10. ¬x = r
. . . . | . . | . 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . . | . . | . 12. D' : Dsys
. . . . | . . | . 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . . | . . | . 14. w : World
. . . . | . . | . 15. p : FairFifo
```

```
. . . . | . . . | . 16. PossibleWorld(D';w)
. . . . | . . . | . 17. vartype(i;r) ⊆r 𝔹
. . . . | . . . | . 18. ∀e:E.
. . . . | . . . | . |   loc(e) = i ⇒ first(e) ⇒ (r when e) = false2
. . . . | . . . | . 19. ∀e:E.
. . . . | . . . | . |   loc(e) = i
. . . . | . . . | . |   ⇒
. . . . | . . . | . |   ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . . | . . . | . 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v)
. . . . | . . . | . |   ∨2 r)
. . . . | . . . | . |   ∈ Dsys
. . . . | . . . | . 21. vartype(i;r) ⊆r 𝔹
. . . . | . . . | . 22. vartype(i;x) ⊆r A
. . . . | . . . | . 23. ∀e:E.
. . . . | . . . | . |   loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . . | . . . | . 24. ∀e:E.
. . . . | . . . | . |   loc(e) = i
. . . . | . . . | . |   ⇒
. . . . | . . . | . |   kind(e) = k
. . . . | . . . | . |   ⇒
. . . . | . . . | . |   (r after e)
. . . . | . . . | . |   =
. . . . | . . . | . |   (P((x when e),val(e)) ∨2 (r when e))
. . . . | . . . | . 25. e' : E
. . . . | . . . | . 26. loc(e') = i
. . . . | . . . | . 27. ¬first(e')
. . . . | . . . | . 28. (r after pred(e'))
. . . . | . . . | . |   ⇒
. . . . | . . . | . |   (∃e@0:E.
. . . . | . . . | . |    (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . . . | . |    & kind(e@0) = k
. . . . | . . . | . |    & P((x when e@0),val(e@0)))
. . . . | . . . | . 29. (r when e') = (r after pred(e'))
. . . . | . . . | . 30. e@0 : E
. . . . | . . . | . 31. (e@0 <loc e')
. . . . | . . . | . 32. kind(e@0) = k
. . . . | . . . | . 33. P((x when e@0),val(e@0))
. . . . | . . . | . 34. (r after pred(e'))
. . . . | . . . | . 35. ¬(r after e')
. . . . | . . . | . ⊢  (r after e')  by Assert (kind(e') = k)
. . . . | . . . | . |
. . . . | . . . | . |\
. . . . | . . . | . | .....assertion .....
. . . . | . . . | . | 1. T : Type
. . . . | . . . | . | 2. A : Type
. . . . | . . . | . | 3. P : A→T→𝔹
. . . . | . . . | . | 4. k : Knd
. . . . | . . . | . | 5. i : Id
. . . . | . . . | . | 6. r : Id
. . . . | . . . | . | 7. x : Id
. . . . | . . . | . | 8. A
. . . . | . . . | . | 9. T
. . . . | . . . | . | 10. ¬x = r
. . . . | . . . | . | 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)))
. . . . | . . . | . | |   ∈ Dsys
```

```
. . . . | . . . | . . | 12. D' : Dsys
. . . . | . . . | . . | 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . . | . . . | . . | 14. w : World
. . . . | . . . | . . | 15. p : FairFifo
. . . . | . . . | . . | 16. PossibleWorld(D';w)
. . . . | . . . | . . | 17. vartype(i;r) ⊆r 𝔹
. . . . | . . . | . . | 18. ∀e:E.
. . . . | . . . | . . | |   loc(e) = i ⇒ first(e) ⇒ (r when e) = false₂
. . . . | . . . | . . | 19. ∀e:E.
. . . . | . . . | . . | |   loc(e) = i
. . . . | . . . | . . | |   ⇒
. . . . | . . . | . . | |   ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . . | . . . | . . | 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P
. . . . | . . . | . . | |                                                (x
. . . . | . . . | . . | |                                                ,v)
. . . . | . . . | . . | |   ∨₂ r)
. . . . | . . . | . . | |   ∈ Dsys
. . . . | . . . | . . | 21. vartype(i;r) ⊆r 𝔹
. . . . | . . . | . . | 22. vartype(i;x) ⊆r A
. . . . | . . . | . . | 23. ∀e:E.
. . . . | . . . | . . | |   loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . . | . . . | . . | 24. ∀e:E.
. . . . | . . . | . . | |   loc(e) = i
. . . . | . . . | . . | |   ⇒
. . . . | . . . | . . | |   kind(e) = k
. . . . | . . . | . . | |   ⇒
. . . . | . . . | . . | |   (r after e)
. . . . | . . . | . . | |   =
. . . . | . . . | . . | |   (P((x when e),val(e)) ∨₂ (r when e))
. . . . | . . . | . . | 25. e' : E
. . . . | . . . | . . | 26. loc(e') = i
. . . . | . . . | . . | 27. ¬first(e')
. . . . | . . . | . . | 28. (r after pred(e'))
. . . . | . . . | . . | |   ⇒
. . . . | . . . | . . | |   (∃e@0:E.
. . . . | . . . | . . | |    (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . . . | . . | |    & kind(e@0) = k
. . . . | . . . | . . | |    & P((x when e@0),val(e@0)))
. . . . | . . . | . . | 29. (r when e') = (r after pred(e'))
. . . . | . . . | . . | 30. e@0 : E
. . . . | . . . | . . | 31. (e@0 <loc e')
. . . . | . . . | . . | 32. kind(e@0) = k
. . . . | . . . | . . | 33. P((x when e@0),val(e@0))
. . . . | . . . | . . | 34. (r after pred(e'))
. . . . | . . . | . . | 35. ¬(r after e')
. . . . | . . . | . . | ⊢  kind(e') = k  by BackThruSomeHyp
. . . . | . . . | . . | |
. . . . | . . . | . . | 1. T : Type
. . . . | . . . | . . | 2. A : Type
. . . . | . . . | . . | 3. P : A→T→𝔹
. . . . | . . . | . . | 4. k : Knd
. . . . | . . . | . . | 5. i : Id
. . . . | . . . | . . | 6. r : Id
. . . . | . . . | . . | 7. x : Id
. . . . | . . . | . . | 8. A
```

```
. . . . | . . | . | 9. T
. . . . | . . | . | 10. ¬x = r
. . . . | . . | . | 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)))
. . . . | . . | . |     ∈ Dsys
. . . . | . . | . | 12. D' : Dsys
. . . . | . . | . | 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . . | . . | . | 14. w : World
. . . . | . . | . | 15. p : FairFifo
. . . . | . . | . | 16. PossibleWorld(D';w)
. . . . | . . | . | 17. vartype(i;r) ⊆r 𝔹
. . . . | . . | . | 18. ∀e:E.
. . . . | . . | . |     loc(e) = i ⇒ first(e) ⇒ (r when e) = false2
. . . . | . . | . | 19. ∀e:E.
. . . . | . . | . |     loc(e) = i
. . . . | . . | . |     ⇒
. . . . | . . | . |     ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . . | . . | . | 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P
. . . . | . . | . |                                                  (x
. . . . | . . | . |                                                  ,v)
. . . . | . . | . |     ∨2 r)
. . . . | . . | . |     ∈ Dsys
. . . . | . . | . | 21. vartype(i;r) ⊆r 𝔹
. . . . | . . | . | 22. vartype(i;x) ⊆r A
. . . . | . . | . | 23. ∀e:E.
. . . . | . . | . |     loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . . | . . | . | 24. ∀e:E.
. . . . | . . | . |     loc(e) = i
. . . . | . . | . |     ⇒
. . . . | . . | . |     kind(e) = k
. . . . | . . | . |     ⇒
. . . . | . . | . |     (r after e)
. . . . | . . | . |     =
. . . . | . . | . |     (P((x when e),val(e)) ∨2 (r when e))
. . . . | . . | . | 25. e' : E
. . . . | . . | . | 26. loc(e') = i
. . . . | . . | . | 27. ¬first(e')
. . . . | . . | . | 28. (r after pred(e'))
. . . . | . . | . |     ⇒
. . . . | . . | . |     (∃e@0:E.
. . . . | . . | . |      (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . . | . |      & kind(e@0) = k
. . . . | . . | . |      & P((x when e@0),val(e@0)))
. . . . | . . | . | 29. (r when e') = (r after pred(e'))
. . . . | . . | . | 30. e@0 : E
. . . . | . . | . | 31. (e@0 <loc e')
. . . . | . . | . | 32. kind(e@0) = k
. . . . | . . | . | 33. P((x when e@0),val(e@0))
. . . . | . . | . | 34. (r after pred(e'))
. . . . | . . | . | 35. ¬(r after e')
. . . . | . . | . | ⊢  ¬(r after e') = (r when e')
. . . . | . . | . |   by MaBoolCase (r after e')
. . . . | . . | . |      THEN

. . . . | . . | . |      MaBoolCase (r when e')
. . . . | . . | . |      THEN
. . . . | . . | . |      MaBoolCase (r after pred(e'))
```

```
. . . . | . . | . |
. . . . | . . | . .\
. . . . | . . | . . 1. T : Type
. . . . | . . | . . 2. A : Type
. . . . | . . | . . 3. P : A→T→𝔹
. . . . | . . | . . 4. k : Knd
. . . . | . . | . . 5. i : Id
. . . . | . . | . . 6. r : Id
. . . . | . . | . . 7. x : Id
. . . . | . . | . . 8. A
. . . . | . . | . . 9. T
. . . . | . . | . . 10. ¬x = r
. . . . | . . | . . 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)))
. . . . | . . | . .     ∈ Dsys
. . . . | . . | . . 12. D' : Dsys
. . . . | . . | . . 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . . | . . | . . 14. w : World
. . . . | . . | . . 15. p : FairFifo
. . . . | . . | . . 16. PossibleWorld(D';w)
. . . . | . . | . . 17. vartype(i;r) ⊆r 𝔹
. . . . | . . | . . 18. ∀e:E.
. . . . | . . | . .     loc(e) = i ⇒ first(e) ⇒ (r when e) = false₂
. . . . | . . | . . 19. ∀e:E.
. . . . | . . | . .     loc(e) = i
. . . . | . . | . .     ⇒
. . . . | . . | . .     ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . . | . . | . . 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P
. . . . | . . | . .                                                       (x
. . . . | . . | . .                                                       ,v)
. . . . | . . | . .     ∨₂ r)
. . . . | . . | . .     ∈ Dsys
. . . . | . . | . . 21. vartype(i;r) ⊆r 𝔹
. . . . | . . | . . 22. vartype(i;x) ⊆r A
. . . . | . . | . . 23. ∀e:E.
. . . . | . . | . .     loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . . | . . | . . 24. ∀e:E.
. . . . | . . | . .     loc(e) = i
. . . . | . . | . .     ⇒
. . . . | . . | . .     kind(e) = k
. . . . | . . | . .     ⇒
. . . . | . . | . .     (r after e)
. . . . | . . | . .     =
. . . . | . . | . .     (P((x when e),val(e)) ∨₂ (r when e))
. . . . | . . | . . 25. e' : E
. . . . | . . | . . 26. loc(e') = i
. . . . | . . | . . 27. ¬first(e')
. . . . | . . | . . 28. (r after pred(e'))
. . . . | . . | . .     ⇒
. . . . | . . | . .     (∃e@0:E.
. . . . | . . | . .      (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . . | . .      & kind(e@0) = k
. . . . | . . | . .      & P((x when e@0),val(e@0)))
. . . . | . . | . . 29. (r when e') = (r after pred(e'))
. . . . | . . | . . 30. e@0 : E
. . . . | . . | . . 31. (e@0 <loc e')
```

```
. . . . | . . | . . 32. kind(e@0) = k
. . . . | . . | . . 33. P((x when e@0),val(e@0))
. . . . | . . | . . 34. (r after pred(e'))
. . . . | . . | . . 35. ¬(r after e')
. . . . | . . | . . 36. kind(e') = k
. . . . | . . | . . ⊢  (r after e')
. . . . | . . | . .   by RW (SweepDnC (HypC 24)) 0 THENA MaAuto
. . . . | . . | . .      THEN
. . . . | . . | . .      MaBoolCase (r when e')
. . . . | . . | . .      THEN
. . . . | . . | . .      MaBoolCase (P((x when e'),val(e')))
. . . . | . . | . .      THEN
. . . . | . . | . .      MaBoolCase (r after pred(e'))
. . . . | . . |
. . . . | . . .\
. . . . | . . . 1. T : Type
. . . . | . . . 2. A : Type
. . . . | . . . 3. P : A→T→𝔹
. . . . | . . . 4. k : Knd
. . . . | . . . 5. i : Id
. . . . | . . . 6. r : Id
. . . . | . . . 7. x : Id
. . . . | . . . 8. A
. . . . | . . . 9. T
. . . . | . . . 10. ¬x = r
. . . . | . . . 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . . | . . . 12. D' : Dsys
. . . . | . . . 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . . | . . . 14. w : World
. . . . | . . . 15. p : FairFifo
. . . . | . . . 16. PossibleWorld(D';w)
. . . . | . . . 17. vartype(i;r) ⊆r 𝔹
. . . . | . . . 18. ∀e:E.
. . . . | . . . |   loc(e) = i ⇒ first(e) ⇒ (r when e) = false₂
. . . . | . . . 19. ∀e:E.
. . . . | . . . |   loc(e) = i
. . . . | . . . |   ⇒
. . . . | . . . |   ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . . | . . . 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v)
. . . . | . . . |   ∨₂ r)
. . . . | . . . |   ∈ Dsys
. . . . | . . . 21. vartype(i;r) ⊆r 𝔹
. . . . | . . . 22. vartype(i;x) ⊆r A
. . . . | . . . 23. ∀e:E.
. . . . | . . . |   loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . . | . . . 24. ∀e:E.
. . . . | . . . |   loc(e) = i
. . . . | . . . |   ⇒
. . . . | . . . |   kind(e) = k
. . . . | . . . |   ⇒
. . . . | . . . |   (r after e) = (P((x when e),val(e)) ∨₂ (r when e))
. . . . | . . . 25. e' : E
. . . . | . . . 26. loc(e') = i
. . . . | . . . 27. ¬first(e')
. . . . | . . . 28. (r after pred(e'))
```

```
. . . . | . . . |   ⇒
. . . . | . . . |   (∃e@0:E.
. . . . | . . . |    (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . . . |    & kind(e@0) = k
. . . . | . . . |    & P((x when e@0),val(e@0)))
. . . . | . . . 29. (r after pred(e')) ⇐ (∃e@0:E.
. . . . | . . . |   (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . . . |   & kind(e@0) = k
. . . . | . . . |   & P((x when e@0),val(e@0)))
. . . . | . . . 30. (r when e') = (r after pred(e'))
. . . . | . . . 31. e@0 : E
. . . . | . . . 32. e@0 = e'
. . . . | . . . 33. kind(e@0) = k
. . . . | . . . 34. P((x when e@0),val(e@0))
. . . . | . . . ⊢  (r after e')  by HypSubst -3 -2
. . . . | . . . |
. . . . | . . . 1. T : Type
. . . . | . . . 2. A : Type
. . . . | . . . 3. P : A→T→𝔹
. . . . | . . . 4. k : Knd
. . . . | . . . 5. i : Id
. . . . | . . . 6. r : Id
. . . . | . . . 7. x : Id
. . . . | . . . 8. A
. . . . | . . . 9. T
. . . . | . . . 10. ¬x = r
. . . . | . . . 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . . | . . . 12. D' : Dsys
. . . . | . . . 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . . | . . . 14. w : World
. . . . | . . . 15. p : FairFifo
. . . . | . . . 16. PossibleWorld(D';w)
. . . . | . . . 17. vartype(i;r) ⊆r 𝔹
. . . . | . . . 18. ∀e:E.
. . . . | . . . |   loc(e) = i ⇒ first(e) ⇒ (r when e) = false2
. . . . | . . . 19. ∀e:E.
. . . . | . . . |   loc(e) = i
. . . . | . . . |   ⇒
. . . . | . . . |   ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . . | . . . 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v)
. . . . | . . . |   ∨2 r)
. . . . | . . . |   ∈ Dsys
. . . . | . . . 21. vartype(i;r) ⊆r 𝔹
. . . . | . . . 22. vartype(i;x) ⊆r A
. . . . | . . . 23. ∀e:E.
. . . . | . . . |   loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . . | . . . 24. ∀e:E.
. . . . | . . . |   loc(e) = i
. . . . | . . . |   ⇒
. . . . | . . . |   kind(e) = k
. . . . | . . . |   ⇒
. . . . | . . . |   (r after e) = (P((x when e),val(e)) ∨2 (r when e))
. . . . | . . . 25. e' : E
. . . . | . . . 26. loc(e') = i
. . . . | . . . 27. ¬first(e')
```

```
. . . . | . . . 28. (r after pred(e'))
. . . . | . . . |   ⇒
. . . . | . . . |   (∃e@0:E.
. . . . | . . . |    (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . . . |    & kind(e@0) = k
. . . . | . . . |    & P((x when e@0),val(e@0)))
. . . . | . . . 29. (r after pred(e')) ⇐ (∃e@0:E.
. . . . | . . . |   (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . . . |   & kind(e@0) = k
. . . . | . . . |   & P((x when e@0),val(e@0)))
. . . . | . . . 30. (r when e') = (r after pred(e'))
. . . . | . . . 31. e@0 : E
. . . . | . . . 32. e@0 = e'
. . . . | . . . 33. kind(e') = k
. . . . | . . . 34. P((x when e@0),val(e@0))
. . . . | . . . ⊢  (r after e')
. . . . | . . . | by ((RW (SweepDnC (HypC 24)) 0)
. . . . | . . . |     THEN
. . . . | . . . |     (RW assert_pushdownC 0))
. . . . | . . . |    THENA
. . . . | . . . |    MaAuto
. . . . | . . . |
. . . . | . . . 1. T : Type
. . . . | . . . 2. A : Type
. . . . | . . . 3. P : A→T→𝔹
. . . . | . . . 4. k : Knd
. . . . | . . . 5. i : Id
. . . . | . . . 6. r : Id
. . . . | . . . 7. x : Id
. . . . | . . . 8. A
. . . . | . . . 9. T
. . . . | . . . 10. ¬x = r
. . . . | . . . 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . . | . . . 12. D' : Dsys
. . . . | . . . 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . . | . . . 14. w : World
. . . . | . . . 15. p : FairFifo
. . . . | . . . 16. PossibleWorld(D';w)
. . . . | . . . 17. vartype(i;r) ⊆r 𝔹
. . . . | . . . 18. ∀e:E.
. . . . | . . . |   loc(e) = i ⇒ first(e) ⇒ (r when e) = false_2
. . . . | . . . 19. ∀e:E.
. . . . | . . . |   loc(e) = i
. . . . | . . . |   ⇒
. . . . | . . . |   ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . . | . . . 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v)
. . . . | . . . |   ∨_2 r)
. . . . | . . . |   ∈ Dsys
. . . . | . . . 21. vartype(i;r) ⊆r 𝔹
. . . . | . . . 22. vartype(i;x) ⊆r A
. . . . | . . . 23. ∀e:E.
. . . . | . . . |   loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . . | . . . 24. ∀e:E.
. . . . | . . . |   loc(e) = i
. . . . | . . . |   ⇒
```

```
. . . . | . . . |   kind(e) = k
. . . . | . . . |   ⇒
. . . . | . . . |   (r after e) = (P((x when e),val(e)) ∨2 (r when e))
. . . . | . . . 25. e' : E
. . . . | . . . 26. loc(e') = i
. . . . | . . . 27. ¬first(e')
. . . . | . . . 28. (r after pred(e'))
. . . . | . . . |   ⇒
. . . . | . . . |   (∃e@0:E.
. . . . | . . . |    (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . . . |    & kind(e@0) = k
. . . . | . . . |    & P((x when e@0),val(e@0)))
. . . . | . . . 29. (r after pred(e')) ⇐ (∃e@0:E.
. . . . | . . . |   (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . . . |   & kind(e@0) = k
. . . . | . . . |   & P((x when e@0),val(e@0)))
. . . . | . . . 30. (r when e') = (r after pred(e'))
. . . . | . . . 31. e@0 : E
. . . . | . . . 32. e@0 = e'
. . . . | . . . 33. kind(e') = k
. . . . | . . . 34. P((x when e@0),val(e@0))
. . . . | . . . ⊢  P((x when e'),val(e')) ∨ (r when e')  by OrLeft
. . . . | . . . |
. . . . | . . . 1. T : Type
. . . . | . . . 2. A : Type
. . . . | . . . 3. P : A→T→𝔹
. . . . | . . . 4. k : Knd
. . . . | . . . 5. i : Id
. . . . | . . . 6. r : Id
. . . . | . . . 7. x : Id
. . . . | . . . 8. A
. . . . | . . . 9. T
. . . . | . . . 10. ¬x = r
. . . . | . . . 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . . | . . . 12. D' : Dsys
. . . . | . . . 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . . | . . . 14. w : World
. . . . | . . . 15. p : FairFifo
. . . . | . . . 16. PossibleWorld(D';w)
. . . . | . . . 17. vartype(i;r) ⊆r 𝔹
. . . . | . . . 18. ∀e:E.
. . . . | . . . |   loc(e) = i ⇒ first(e) ⇒ (r when e) = false2
. . . . | . . . 19. ∀e:E.
. . . . | . . . |   loc(e) = i
. . . . | . . . |   ⇒
. . . . | . . . |   ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . . | . . . 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v)
. . . . | . . . |   ∨2 r)
. . . . | . . . |   ∈ Dsys
. . . . | . . . 21. vartype(i;r) ⊆r 𝔹
. . . . | . . . 22. vartype(i;x) ⊆r A
. . . . | . . . 23. ∀e:E.
. . . . | . . . |   loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . . | . . . 24. ∀e:E.
. . . . | . . . |   loc(e) = i
```

```
. . . . | . . . |   ⇒
. . . . | . . . |   kind(e) = k
. . . . | . . . |   ⇒
. . . . | . . . |   (r after e) = (P((x when e),val(e)) ∨₂ (r when e))
. . . . | . . . 25. e' : E
. . . . | . . . 26. loc(e') = i
. . . . | . . . 27. ¬first(e')
. . . . | . . . 28. (r after pred(e'))
. . . . | . . . |   ⇒
. . . . | . . . |   (∃e@0:E.
. . . . | . . . |    (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . . . |    & kind(e@0) = k
. . . . | . . . |    & P((x when e@0),val(e@0)))
. . . . | . . . 29. (r after pred(e')) ⇐ (∃e@0:E.
. . . . | . . . |   (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . . . |   & kind(e@0) = k
. . . . | . . . |   & P((x when e@0),val(e@0)))
. . . . | . . . 30. (r when e') = (r after pred(e'))
. . . . | . . . 31. e@0 : E
. . . . | . . . 32. e@0 = e'
. . . . | . . . 33. kind(e') = k
. . . . | . . . 34. P((x when e@0),val(e@0))

. . . . | . . . ⊢  P((x when e'),val(e'))
. . . . | . . . | by StrongRevHypSubst -3 0 THEN MaAuto
. . . . | . . . |
. . . . | . . . 1. T : Type
. . . . | . . . 2. A : Type
. . . . | . . . 3. P : A→T→𝔹
. . . . | . . . 4. k : Knd
. . . . | . . . 5. i : Id
. . . . | . . . 6. r : Id
. . . . | . . . 7. x : Id
. . . . | . . . 8. A
. . . . | . . . 9. T
. . . . | . . . 10. ¬x = r
. . . . | . . . 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . . | . . . 12. D' : Dsys
. . . . | . . . 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . . | . . . 14. w : World
. . . . | . . . 15. p : FairFifo
. . . . | . . . 16. PossibleWorld(D';w)
. . . . | . . . 17. vartype(i;r) ⊆r 𝔹
. . . . | . . . 18. ∀e:E.
. . . . | . . . |   loc(e) = i ⇒ first(e) ⇒ (r when e) = false₂
. . . . | . . . 19. ∀e:E.
. . . . | . . . |   loc(e) = i
. . . . | . . . |   ⇒
. . . . | . . . |   ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . . | . . . 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v)
. . . . | . . . |   ∨₂ r)
. . . . | . . . |   ∈ Dsys
. . . . | . . . 21. vartype(i;r) ⊆r 𝔹
. . . . | . . . 22. vartype(i;x) ⊆r A
. . . . | . . . 23. ∀e:E.
. . . . | . . . |   loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
```

```
. . . . | . . . 24. ∀e:E.
. . . . | . . . |   loc(e) = i
. . . . | . . . |   ⇒
. . . . | . . . |   kind(e) = k
. . . . | . . . |   ⇒
. . . . | . . . |   (r after e) = (P((x when e),val(e)) ∨2 (r when e))
. . . . | . . . 25. e' : E
. . . . | . . . 26. loc(e') = i
. . . . | . . . 27. ¬first(e')
. . . . | . . . 28. (r after pred(e'))
. . . . | . . . |   ⇒
. . . . | . . . |   (∃e@0:E.
. . . . | . . . |    (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . . . |    & kind(e@0) = k
. . . . | . . . |    & P((x when e@0),val(e@0)))
. . . . | . . . 29. (r after pred(e')) ⇐ (∃e@0:E.
. . . . | . . . |   (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . . . |   & kind(e@0) = k
. . . . | . . . |   & P((x when e@0),val(e@0)))
. . . . | . . . 30. (r when e') = (r after pred(e'))
. . . . | . . . 31. e@0 : E
. . . . | . . . 32. e@0 = e'
. . . . | . . . 33. kind(e') = k
. . . . | . . . 34. P((x when e@0),val(e@0))
. . . . | . . . 35. z : E
. . . . | . . . 36. z = e@0
. . . . | . . . ⊢  val(z) ∈ T  by DoSubsume THEN HypSubst -1 0
. . . . | . . . |
. . . . | . . . 1. T : Type
. . . . | . . . 2. A : Type
. . . . | . . . 3. P : A→T→𝔹
. . . . | . . . 4. k : Knd
. . . . | . . . 5. i : Id
. . . . | . . . 6. r : Id
. . . . | . . . 7. x : Id
. . . . | . . . 8. A
. . . . | . . . 9. T
. . . . | . . . 10. ¬x = r
. . . . | . . . 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . . | . . . 12. D' : Dsys
. . . . | . . . 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . . | . . . 14. w : World
. . . . | . . . 15. p : FairFifo
. . . . | . . . 16. PossibleWorld(D';w)
. . . . | . . . 17. vartype(i;r) ⊆r 𝔹
. . . . | . . . 18. ∀e:E.
. . . . | . . .     loc(e) = i ⇒ first(e) ⇒ (r when e) = false2
. . . . | . . . 19. ∀e:E.
. . . . | . . .     loc(e) = i
. . . . | . . .     ⇒
. . . . | . . .     ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . . | . . . 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v)
. . . . | . . .     ∨2 r)
. . . . | . . .     ∈ Dsys
. . . . | . . . 21. vartype(i;r) ⊆r 𝔹
```

```
. . . . | . . . 22. vartype(i;x) ⊆r A
. . . . | . . . 23. ∀e:E.
. . . . | . . .     loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . . | . . . 24. ∀e:E.
. . . . | . . .     loc(e) = i
. . . . | . . .     ⇒
. . . . | . . .     kind(e) = k
. . . . | . . .     ⇒
. . . . | . . .     (r after e) = (P((x when e),val(e)) ∨2 (r when e))
. . . . | . . . 25. e' : E
. . . . | . . . 26. loc(e') = i
. . . . | . . . 27. ¬first(e')
. . . . | . . . 28. (r after pred(e'))
. . . . | . . .     ⇒
. . . . | . . .     (∃e@0:E.
. . . . | . . .      (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . . .      & kind(e@0) = k
. . . . | . . .      & P((x when e@0),val(e@0)))
. . . . | . . . 29. (r after pred(e')) ⇐ (∃e@0:E.
. . . . | . . .     (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . | . . .     & kind(e@0) = k
. . . . | . . .     & P((x when e@0),val(e@0)))
. . . . | . . . 30. (r when e') = (r after pred(e'))
. . . . | . . . 31. e@0 : E
. . . . | . . . 32. e@0 = e'
. . . . | . . . 33. kind(e') = k
. . . . | . . . 34. P((x when e@0),val(e@0))
. . . . | . . . 35. z : E
. . . . | . . . 36. z = e@0
. . . . | . . . ⊢  valtype(e@0) ⊆r T  by BackThruSomeHyp THEN MaAuto
. . . . |
. . . . .\
. . . . . .....wf .....
. . . . . 1. T : Type
. . . . . 2. A : Type
. . . . . 3. P : A→T→𝔹
. . . . . 4. k : Knd
. . . . . 5. i : Id
. . . . . 6. r : Id
. . . . . 7. x : Id
. . . . . 8. A
. . . . . 9. T
. . . . . 10. ¬x = r
. . . . . 11. (λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x))) ∈ Dsys
. . . . . 12. D' : Dsys
. . . . . 13. λloc.⊕(recognizer1(loc;T;A;P;k;i;r;x)) ⊆ D'
. . . . . 14. w : World
. . . . . 15. p : FairFifo
. . . . . 16. PossibleWorld(D';w)
. . . . . 17. vartype(i;r) ⊆r 𝔹
. . . . . 18. ∀e:E. loc(e) = i ⇒ first(e) ⇒ (r when e) = false2
. . . . . 19. ∀e:E.
. . . . .     loc(e) = i ⇒ ¬(r after e) = (r when e) ⇒ kind(e) = k
. . . . . 20. @i: ma-single-effect1(r;𝔹;x;A;k;T;λr,x,v. P(x,v) ∨2 r)
. . . . .     ∈ Dsys
```

```
. . . . . 21. vartype(i;r) ⊆r 𝔹
. . . . . 22. vartype(i;x) ⊆r A
. . . . . 23. ∀e:E. loc(e) = i ⇒ kind(e) = k ⇒ (valtype(e) ⊆r T)
. . . . . 24. ∀e:E.
. . . . .     loc(e) = i
. . . . .     ⇒
. . . . .     kind(e) = k
. . . . .     ⇒
. . . . .     (r after e) = (P((x when e),val(e)) ∨2 (r when e))
. . . . . 25. e' : E
. . . . . 26. loc(e') = i
. . . . . 27. ¬first(e')
. . . . . ⊢  ((r after pred(e'))
. . . . .     ⟺
. . . . .     (∃e@0:E.
. . . . .      (e@0 <loc pred(e')) ∨ e@0 = pred(e')
. . . . .      & kind(e@0) = k
. . . . .      & P((x when e@0),val(e@0))))
. . . . .    ∈ Prop  by MaAuto
END
```