```
Rl_iff_Rg  THM

⊢  ∀R:(A*→A*→Prop).
|  (EquivRel x,y:A*. x R y) ⇒
|  (∀x,y,z:A*. (x R y) ⇒ ((z @ x) R (z @ y))) ⇒
|  (∀g:((x,y:A*//(x R y))→𝔹), L:LangOver(A).
|   (∀l:A*. L(l) ⟺ g(l)) ⇒
|   (∀x,y:A*. (x L-induced Equiv y) ⟺ (x Rg y)))  by UnivCD
|
|\
| 1. A : Type
| 2. R : A*→A*→Prop
| 3. EquivRel x,y:A*. x R y
| 4. ∀x,y,z:A*. (x R y) ⇒ ((z @ x) R (z @ y))
| 5. g : (x,y:A*//(x R y))→𝔹
| 6. L : LangOver(A)
| 7. ∀l:A*. L(l) ⟺ g(l)
| 8. x : A*
| 9. y : A*
| ⊢  (x L-induced Equiv y) ⟺ (x Rg y)
| | by (Unfolds [‘lang_rel‘;‘lquo_rel‘] 0)
| |    THEN
| |    (Unfold ‘mn_quo_append‘ 0)
| |    THEN
| |    (Reduce 0)
| |
| ⊢  (∀z:A*. L(z @ x) ⟺ L(z @ y)) ⟺ (∀z:A*. g(z @ x) ⟺ g(z @ y))
| | by (Assert (L ∈ A*→Prop)) THEN (RWO "7" 0)
| |
| |\
| | .....assertion .....
| | ⊢  L ∈ A*→Prop  by Unfold ‘languages‘ 6
| |
| .\
| . 10. L ∈ A*→Prop
| . ⊢  (∀z:A*. g(z @ x) ⟺ g(z @ y)) ⟺ (∀z:A*. g(z @ x) ⟺ g(z @ y))
| .   by (Analyze 0) THEN (Analyze 0) THEN Trivial
|
.\
. 1. A : Type
. 2. R : A*→A*→Prop
. 3. EquivRel x,y:A*. x R y
. 4. ∀x,y,z:A*. (x R y) ⇒ ((z @ x) R (z @ y))
. 5. (x,y:A*//(x R y))→𝔹
. 6. L : LangOver(A)
. 7. A*
. ⊢  L ∈ A*→Prop  by Unfold ‘languages‘ 6
END
```