```
lang_eq_imp_quo_eq  THM

⊢  ∀Alph,S1,S2:Type, A1:Automata(Alph;S1), A2:Automata(Alph;S2).
|  LangOf(A1) = LangOf(A2) ⇒
|  x,y:Alph*//(x LangOf(A1)-induced Equiv y)
|  =
|  x,y:Alph*//(x LangOf(A2)-induced Equiv y)  by UnivCD
|
1. Alph : Type
2. S1 : Type
3. S2 : Type
4. A1 : Automata(Alph;S1)
5. A2 : Automata(Alph;S2)
6. LangOf(A1) = LangOf(A2)
⊢  x,y:Alph*//(x LangOf(A1)-induced Equiv y)
|  =
|  x,y:Alph*//(x LangOf(A2)-induced Equiv y)
| by Inst
|    Thm* ∀L:LangOver(A). EquivRel x,y:A*. x L-induced Equiv y
|    {lang_rel_equi}
|    [Alph;LangOf(A1)]
|    THEN
|    Inst
|    Thm* ∀L:LangOver(A). EquivRel x,y:A*. x L-induced Equiv y
|    {lang_rel_equi}
|    [Alph;LangOf(A2)]
|
7. EquivRel x,y:Alph*. x LangOf(A1)-induced Equiv y
8. EquivRel x,y:Alph*. x LangOf(A2)-induced Equiv y
⊢
| by Refine
|    RULE H  ⊢ x,y:A//E = u,v:B//F ∈ Type{i}
|
|           BY quotientEquality r; s; w
|
|           H ⊢ x,y:A//E ∈ Type{i}
|           H ⊢ u,v:B//F ∈ Type{i}
|           H ⊢ A = B ∈ Type{i}
|           H, w:(A = B ∈ Type{i}), r:A, s:A
|              ⊢ E[r,s/x,y] ⟺ F[r,s/u,v]
|    {quotientEquality}
|    RuleArgs:[x:v ; y:v ; z:v]
|    THEN
|    Try (Complete Auto)
|
9. Alph* = Alph*
10. x : Alph*
11. y : Alph*
⊢  (x LangOf(A1)-induced Equiv y) ⟺ (x LangOf(A2)-induced Equiv y)
| by Analyze THEN Analyze
|
|\
| 12. x LangOf(A1)-induced Equiv y
| ⊢  x LangOf(A2)-induced Equiv y
| | by Unfold ‘lang_rel‘ 0 THEN Reduce 0
```

```
| |    THEN
| |    Unfold `lang_rel` -1 THEN Reduce -1
| |
| 12. ∀z:Alph*. LangOf(A1)(z @ x) ⟺ LangOf(A1)(z @ y)
| ⊢  ∀z:Alph*. LangOf(A2)(z @ x) ⟺ LangOf(A2)(z @ y)
| | by SimilarTo -1
| |
| |\
| | 12. z1 : Alph*
| | 13. LangOf(A1)(z1 @ x) ⇒ LangOf(A1)(z1 @ y)
| | ⊢  LangOf(A2)(z1 @ x) ⇒ LangOf(A2)(z1 @ y)  by SimilarTo -1
| | |
| | |\
| | | .....antecedent .....
| | | 13. LangOf(A2)(z1 @ x)
| | | ⊢  LangOf(A1)(z1 @ x)  by BackThru: 6
| | |
| | |\
| | | 13. LangOf(A2)(z1 @ x)
| | | 14. LangOf(A1)(z1 @ y)
| | | ⊢  LangOf(A2)(z1 @ y)  by BackThru: 6
| | |
| | .\
| | . ⊢  LangOf(A2) ∈ Alph*→Prop  by ←Def of LangOver(<Alph>)
| |
| .\
| . 12. z1 : Alph*
| . 13. (LangOf(A1)(z1 @ x)) ⇐ (LangOf(A1)(z1 @ y))
| . ⊢  (LangOf(A2)(z1 @ x)) ⇐ (LangOf(A2)(z1 @ y))
| .   by {like the other branch }
| .      SimilarTo -1 THENP BackThru: 6
| .      THENL
| .      [BackThru: 6;←Def of LangOver(<Alph>)]
|
.\
. 12. x LangOf(A2)-induced Equiv y
. ⊢  x LangOf(A1)-induced Equiv y
.   by {like the other branch }
.      Unfold `lang_rel` 0 THEN Reduce 0
.      THEN
.      Unfold `lang_rel` -1 THEN Reduce -1
.      THEN
.      SimilarTo -1
.      THENL
.      [SimilarTo -1
.      ;SimilarTo -1 THENP BackThru: 6
.       THENL

.       [BackThru: 6;←Def of LangOver(<Alph>)]]
.      THENP
.      BackThru: 6
.      THENL
.      [BackThru: 6;←Def of LangOver(<Alph>)]
END
```