```
homo_is_inj  THM

⊢  ∀Auto:Automata(Alph;St), c:(St→Alph*).
|  (∀q:St. (Result(Auto)c(q)) = q)
|  & Fin(Alph)
|  & Fin(St)
|  & (St ∼ (x,y:Alph*//(x LangOf(Auto)-induced Equiv y)))
|  ⇒
|  Inj(St; x,y:Alph*//(x LangOf(Auto)-induced Equiv y); c)
| by Analyze 0 THEN Analyze 0 THEN Analyze 0
|
1. Alph : Type
2. St : Type
3. Auto : Automata(Alph;St)
⊢  ∀c:(St→Alph*).
|  (∀q:St. (Result(Auto)c(q)) = q)
|  & Fin(Alph)
|  & Fin(St)
|  & (St ∼ (x,y:Alph*//(x LangOf(Auto)-induced Equiv y)))
|  ⇒
|  Inj(St; x,y:Alph*//(x LangOf(Auto)-induced Equiv y); c)
| by Inst
|    Thm* ∀L:LangOver(A). EquivRel x,y:A*. x L-induced Equiv y
|    {lang_rel_equi}
|    [Alph;LangOf(Auto)]
|
4. EquivRel x,y:Alph*. x LangOf(Auto)-induced Equiv y
⊢   by Auto
|
5. c : St→Alph*
6. ∀q:St. (Result(Auto)c(q)) = q
7. Fin(Alph)
8. Fin(St)
9. St ∼ (x,y:Alph*//(x LangOf(Auto)-induced Equiv y))
⊢  Inj(St; x,y:Alph*//(x LangOf(Auto)-induced Equiv y); c)
| by BackThru
|    Thm* ∀f:(A→B). (A ∼ B) & Fin(B) ⇒ Surj(A; B; f) ⇒ Inj(A; B; f)
|    {surj_is_inj_gen}
|
|\
| ⊢  (St ∼ (x,y:Alph*//(x LangOf(Auto)-induced Equiv y)))
| |  & Fin(x,y:Alph*//(x LangOf(Auto)-induced Equiv y))  by Analyze 0
| |
| |\
| | ⊢  St ∼ (x,y:Alph*//(x LangOf(Auto)-induced Equiv y))  by Auto
| |
| .\
| . ⊢  Fin(x,y:Alph*//(x LangOf(Auto)-induced Equiv y))
| .   by BackThru
| .      Thm* ∀Auto:Automata(Alph;St).
| .           Fin(Alph) & Fin(St) ⇒
| .           Fin(x,y:Alph*//(x LangOf(Auto)-induced Equiv y))
| .      {mn_13}
|
.\
```

```
. ⊢  Surj(St; x,y:Alph*//(x LangOf(Auto)-induced Equiv y); c)
. | by BackThru
. |    Thm* ∀Auto:Automata(Alph;St), c:(St→Alph*).
. |          (∀q:St. (Result(Auto)c(q)) = q) & Fin(Alph) & Fin(St) ⇒
. |          Surj(St; x,y:Alph*//(x LangOf(Auto)-induced Equiv y); c)
. |    {homo_is_surj}
. |
. 10. q : St
. ⊢  (Result(Auto)c(q)) = q  by BackThru 6
END
```