

Oct 24, 16 7:17

ImpConstFold.v

Page 1/2

```

Require Import List.
Require Import String.
Require Import ZArith.

Open Scope list_scope.
Open Scope string_scope.
Open Scope Z_scope.

Require Import StructTactics.
Require Import ImpSyntax.
Require Import ImpCommon.
Require Import ImpExprTransf.
Require Import ImpInterpNock.

Module I := ImpInterpNock.

Definition cfold_aux (e : expr) : expr :=
  match e with
  | Eop1 op (Eval v) =>
    Eval (I.interp_op1 op v)
  | Eop2 op (Eval v1) (Eval v2) =>
    Eval (I.interp_op2 op v1 v2)
  | _ => e
  end.

Definition cfold_e : expr -> expr :=
  transf_e cfold_aux.

Definition cfold (p : prog) : prog :=
  transf_p cfold_e p.

(*
Fixpoint cfold_e (e : expr) : expr :=
  match e with
  | Eval v => Eval v
  | Evar x => Evar x
  | Eop1 op e1 =>
    match cfold_e e1 with
    | Eval v =>
      Eval (I.interp_op1 op v)
    | e1' =>
      Eop1 op e1'
    end
  | Eop2 op e1 e2 =>
    match op, cfold_e e1, cfold_e e2 with
    | _, Eval v1, Eval v2 =>
      Eval (I.interp_op2 op v1 v2)
    | Oadd, e', Eval (Vint 0)
    | Oadd, Eval (Vint 0), e'
    | Osub, e', Eval (Vint 0)
    | Omul, e', Eval (Vint 1)
    | Omul, Eval (Vint 1), e'
    | Odiv, e', Eval (Vint 1) =>
      e'
    | Omul, e', Eval (Vint 0)
    | Omul, Eval (Vint 0), e'
    | Odiv, e', Eval (Vint 0) =>
      Eval (Vint 0)
    | Odiv, e1', e2' =>
      if expr_eq_dec e1' e2' then
        Eval (Vint 1)
      else
        Eop2 Odiv e1' e2'
    | Omod, e1', Eval (Vint 0)
    | Omod, e1', Eval (Vint 1) =>
      Eval (Vint 0)
    | Omod, e1', e2' =>
      if expr_eq_dec e1' e2' then
        Eval (Vint 0)

```

Oct 24, 16 7:17

ImpConstFold.v

Page 2/2

```

      else
        Eop2 Odiv e1' e2'
    | Oeq, e1', e2' =>
      if expr_eq_dec e1' e2' then
        Eval (Vbool true)
      else
        Eop2 Oeq e1' e2'
    | Olt, e1', e2' =>
      if expr_eq_dec e1' e2' then
        Eval (Vbool false)
      else
        Eop2 Ole e1' e2'
    | Ole, e1', e2' =>
      if expr_eq_dec e1' e2' then
        Eval (Vbool true)
      else
        Eop2 Ole e1' e2'
    | Oconj, e', Eval (Vbool true)
    | Oconj, Eval (Vbool true), e'
    | Odisj, e', Eval (Vbool false)
    | Odisj, Eval (Vbool false), e' =>
      e'
    | Oconj, e', Eval (Vbool false)
    | Oconj, Eval (Vbool false), e' =>
      Eval (Vbool false)
    | Odisj, e', Eval (Vbool true)
    | Odisj, Eval (Vbool true), e' =>
      Eval (Vbool true)
    | _, e1', e2' =>
      Eop2 op e1' e2'
    end
  | Elen e1 =>
    Elen (cfold_e e1)
  | Eidx e1 e2 =>
    Eidx (cfold_e e1) (cfold_e e2)
  end.
*)

```