```
theory Diff_Arr_Lookup
  imports Diff_Arr_Rel Master_Assn
begin

context
begin

type_synonym 'a diff_arr = "'a cell ref"

qualified partial_function (heap) lookup ::
  "('a::heap) diff_arr ⇒ nat ⇒ 'a Heap"
where
  "lookup diff_arr i = do {
      cell ← !diff_arr;
      case cell of
        Array array             ⇒
          Array.nth array i
      | Upd i' value diff_arr' ⇒
          if i = i'
          then return value
          else lookup diff_arr' i
  }"

lemma lookup [sep_heap_rules]:
  "<master_assn t * ↑(t ⊢ xs ∼ diff_arr ∧ i < List.length xs)>
     lookup diff_arr i
   <λr. master_assn t * ↑(r = xs!i)>"
  sorry

end

end
```