# Sample Report Using EmitTeX Macros

**Shiu-Kai Chin**

**15 February 2017**

## Abstract

We demonstrate using the EmitTeX structure functions in HOL to typeset HOL terms, types, theorems, and theories. We use *example1* theory as our example theory to print.

**Acknowledgments:** We gratefully acknowledge the hard work, trust, and dedication of our past students in the Syracuse University Cyber Engineering Semester and the Air Force Research Laboratory's Advanced Course (ACE) in Engineering Cybersecurity Boot Camp. They bridged dreams and reality.

# Contents

## Chapter 1

# Executive Summary

**All requirements for this project are satisfied**. In particular, we prove the example theorem, pretty print the HOL theory, and make use of the *EmitTeX* structure to typeset HOL theorems in this report.

The following theorems are proved and their corresponding LaTeX macros used in this report.

[demoTheorem]

$\vdash \forall p\ q.\ p \Rightarrow (p \Rightarrow q) \Rightarrow q$

[prob1Theorem]

$\vdash \forall p\ q.\ p \Rightarrow (p \Rightarrow q) \Rightarrow q$

Chapter 2

# Proof of prob1Theorem

## 2.1 Problem Statement

Our task is to prove the theorem

$\vdash \forall p\ q.\ p \Rightarrow (p \Rightarrow q) \Rightarrow q$

Here it is again. $\vdash \forall p\ q.\ p \Rightarrow (p \Rightarrow q) \Rightarrow q$

## 2.2 HOL Code Proving prob1Theorem

```
val prob1Theorem =
let
  val th1 = ASSUME ‘‘p:bool‘‘
  val th2 = ASSUME ‘‘p ==> q‘‘
  val th3 = MP th2 th1
  val terma = hd (hyp th2)
  val th4 = DISCH terma th3
  val termb = hd (hyp th1)
  val th5 = DISCH termb th4
in
  GENL [‘‘p:bool‘‘,‘‘q:bool‘‘] th5
end
```

## 2.3 Session Transcript

1

```
> val prob1Theorem =
let
  val th1 = ASSUME ‘‘p:bool‘‘
  val th2 = ASSUME ‘‘p ==> q‘‘
  val th3 = MP th2 th1
  val terma = hd (hyp th2)
  val th4 = DISCH terma th3
  val termb = hd (hyp th1)
  val th5 = DISCH termb th4
in
  GENL [‘‘p:bool‘‘,‘‘q:bool‘‘] th5
end;;;;
# # # # # # # # # # # val prob1Theorem =
   |- !p q. p ==> (p ==> q) ==> q:
   thm
```

Chapter 3

# Proof of demoTheorem

## 3.1 Problem Statement

Our task is to prove the following theorem using PROVE.

$\vdash \forall p\ q.\ p \Rightarrow (p \Rightarrow q) \Rightarrow q$

$\vdash \forall p\ q.\ p \Rightarrow (p \Rightarrow q) \Rightarrow q$

## 3.2 HOL Code Proving demoTheorem

```
val demoTheorem = PROVE [] (concl prob1Theorem);
```

## 3.3 Session Transcript

2

```
> val demoTheorem = PROVE [] (concl prob1Theorem);
Meson search level: ....
val demoTheorem =
   |- !p q. p ==> (p ==> q) ==> q:
   thm
```

## Appendix A

# Source Code for example1Script

The following code is from *example1Script.sml*, which is located in a different subdirectory than this file.

```
(***************************************************************************)
(* A first example showing how to create a HOL script file to create a     *)
(* HOL theory, which allows us to name and save theorems we prove for later *)
(* use.                                                                     *)
(* Author: Shiu-Kai Chin                                                    *)
(***************************************************************************)

(***************************************************************************)
(* All HOL script files are ML modules, so we need to declare the file     *)
(* example1Script as an ML structure.  Do this with the "structure: command *)
(* as the very first executable line.  The very last executable line is "end" *)
(***************************************************************************)

structure example1Script = struct

(***************************************************************************)
(* Note: everything after new_theory must be part of a val assignment, when *)
(* using Holmake.  Otherwise, there will be compilation errors. If you don't *)
(* want to assign an expression to a name, just use "val _ =<expression>"   *)
(* The "_" indicates that we don't want to have a name.                     *)
(***************************************************************************)
open HolKernel Parse boolLib bossLib;

val _ = new_theory "example1";



(***************************************************************************)
(* This theorem was proved as part of forwardProofExample.sml               *)
(***************************************************************************)
val prob1Theorem =
let
  val th1 = ASSUME ``p:bool``
  val th2 = ASSUME ``p ==> q``
  val th3 = MP th2 th1
  val terma = hd (hyp th2)
  val th4 = DISCH terma th3
  val termb = hd (hyp th1)
  val th5 = DISCH termb th4
in
  GENL [``p:bool``,``q:bool``] th5
end
```

```
(******************************************************************************)
(* If we want to save prob1Theorem as part of example1Theory, we need to       *)
(* explicitly save it.                                                         *)
(******************************************************************************)
val _ = save_thm("prob1Theorem",prob1Theorem);



(******************************************************************************)
(* Another theorem we proved as part of forwardProofExample.sml                *)
(******************************************************************************)
val demoTheorem = PROVE [] (concl prob1Theorem);
(******************************************************************************)
(* If we want to save prob1Theorem as part of example1Theory, we need to       *)
(* explicitly save it.                                                         *)
(******************************************************************************)
val _ = save_thm("demoTheorem",demoTheorem);

val _ = export_theory();




end (* structure *)
```