



Pipit on the Post: Proving Pre- and Post-Conditions of Reactive Systems (Artifact)

Amos Robinson ✉ 

Sydney, Australia

Alex Potanin ✉ 

Australian National University, Canberra, Australia

Abstract

Synchronous languages such as Lustre and Scade are used to implement safety-critical control systems; proving such programs correct and having the proved properties apply to the compiled code is therefore equally critical. We introduce Pipit, a small synchronous language embedded in F^* , designed for verifying control systems and executing them in real-time. Pipit includes a verified transla-

tion to transition systems; by reusing F^* 's existing proof automation, certain safety properties can be automatically proved by k-induction on the transition system. Pipit can also generate executable code in a subset of F^* which is suitable for compilation and real-time execution on embedded devices. The executable code is deterministic and total and preserves the semantics of the original program.

2012 ACM Subject Classification Computer systems organization → Real-time languages; Theory of computation → Program verification; Software and its engineering → Specialized application languages

Keywords and phrases Lustre, streaming, reactive, verification

Digital Object Identifier 10.4230/DARTS.10.2.19

Related Article Amos Robinson and Alex Potanin, “Pipit on the Post: Proving Pre- and Post-Conditions of Reactive Systems”, in 38th European Conference on Object-Oriented Programming (ECOOP 2024), LIPIcs, Vol. 313, pp. 34:1–34:28, 2024.

<https://doi.org/10.4230/LIPIcs.ECOOP.2024.34>

Related Conference 38th European Conference on Object-Oriented Programming (ECOOP 2024), September 16–20, 2024, Vienna, Austria

Evaluation Policy The artifact has been evaluated as described in the ECOOP 2024 Call for Artifacts and the ACM Artifact Review and Badging Policy.

1 Scope

This artifact includes the mechanised proofs of the theorems stated in the paper, and the implementation of the language itself. It also includes the network driver example (TTCAN) used in the paper.

2 Content

The artifact package includes:

- the implementation of Pipit itself (code);
- all theorems mentioned in the paper are mechanised in F^* ;
- the implementation of the high-level logic of the TTCAN network driver;
- the proofs of the TTCAN network driver.

The artifact also contains the specific versions of F^* and Karamel used to implement Pipit.



© Amos Robinson and Alex Potanin;
licensed under Creative Commons License CC-BY 4.0
Dagstuhl Artifacts Series, Vol. 10, Issue 2, Artifact No. 19, pp. 19:1–19:2



DAGSTUHL
ARTIFACTS SERIES
Schloss Dagstuhl – Leibniz-Zentrum für Informatik,
Dagstuhl Publishing, Germany



19:2 Pipit on the Post: Proving Pre- and Post-Conditions of Reactive Systems (Artifact)

3 Getting the artifact

The artifact endorsed by the Artifact Evaluation Committee is available free of charge on the Dagstuhl Research Online Publication Server (DROPS). In addition, the artifact is also available at: <https://github.com/songlarknet/pipit>.

4 Tested platforms

Tested on Linux and MacOS.

5 License

The artifact is available under license Apache-2.0.

6 MD5 sum of the artifact

d189cb98071ea28d01ac2f98cb52bf42

7 Size of the artifact

10.7 MiB