

## ABSTRACT

*Currently the Digital Systems Laboratory of the Department of Electrical and Information Engineering, Faculty of Engineering, Universitas Gadjah Mada (DEIE FE UGM) is developing a 32-bit embedded processor designed by using SystemC Model whose processor architecture is the General Purpose Processor (GPP), named GAMA32. GPP is a processor architecture that capable of executing standard instruction-sets such as other embedded processor. However, the development phase of GAMA 32 arrived at the problem of the absence of a compiler capable of converting high-level programming language into assembly language which would be converted into machine language by the GAMA32 assembler.*

*To solve the above problems, a simple compiler was developed that was able to handle variable declarations and simple patterns in high-level programming languages. A Python RPLY library is used to accommodate the function of creating a lexer and parser to then generate an Abstract Syntax Tree (AST) in JSON format. Next, AST is used as an input for the code generator. Code generator will produce assembly language based on the syntactic pattern in C programming language in the form of AST which is currently being accommodated by the code generator is the variable declaration, if, and ifelse pattern. The test results show that the code generator developed is able to produce assembly language in accordance with the input AST, register addressing for variables is correct and as expected.*

**Keywords :** *embedded processor, GAMA32, code generator, assembly language, library RPLY, Python*

## INTISARI

Saat ini Laboratorium Sistem Digital Departemen Teknik Elektro dan Teknologi Informasi Fakultas Teknik Universitas Gadjah Mada (DTETI FT UGM) tengah mengembangkan sebuah *embedded processor* 32-bit yang didesain dengan menggunakan *SystemC Model* yang arsitektur prosesornya adalah *General Purpose Processor* (GPP) dan diberi nama GAMA32. GPP adalah arsitektur prosesor yang mampu menjalankan *instruction-set* standar seperti *embedded processor* lainnya. Akan tetapi tahap pengembangan GAMA32 ini sampai pada masalah belum adanya *compiler* yang mampu mengubah bahasa pemrograman level tinggi menjadi bahasa *assembly* yang kemudian akan diubah menjadi bahasa mesin oleh *assembler* GAMA32.

Untuk menyelesaikan permasalahan di atas maka dilakukan pengembangan *compiler* sederhana yang mampu menangani deklarasi variabel dan pola-pola sederhana dalam pemrograman bahasa level tinggi. *Library* RPLY Python digunakan untuk mengakomodasi fungsi dalam membuat *lexer* dan *parser* untuk lalu menghasilkan *Abstract Syntax Tree* (AST) dalam format JSON. AST ini lalu dijadikan masukan untuk *code generator*. *Code generator* inilah yang akan menghasilkan bahasa *assembly* berdasarkan pola sintaksis dalam bahasa pemrograman C dalam bentuk AST yang sementara ini diakomodasi oleh *code generator* adalah pola deklarasi variabel, *if*, dan *if else*. Hasil pengujian menunjukkan bahwa *code generator* yang dikembangkan mampu menghasilkan bahasa *assembly* sesuai dengan masukan AST, pengalamatan *register* untuk variabel sudah tepat, dan sesuai yang diharapkan.

**Kata kunci** – *embedded processor*, GAMA32, *code generator*, bahasa *assembly*, *library* RPLY, Python