

**LEMBAR PENGESAHAN
SKRIPSI**

**RANCANGAN CATU DAYA CADANGAN SRAM
(Static Random Access Memory) PADA Z80 TRAINER**

telah dipertahankan di depan Dewan penguji pada:

Hari, tanggal : Rabu, 8 Juli 2020
Waktu : 09.30 – 10.30 WITA

Dewan Penguji

1. Syahrir Abdussamad, ST., MT
NIP.19750624 200501 1 003
2. Iskandar Zulkarnain Nasibu, S.Pd., M.Eng
NIP.19701104 200112 1 001
3. Bambang Panji Asmara, ST., MT
NIP.19700405 200912 1 001
4. Amirudin Y. Dako, ST., M.Eng
NIP.19741003 200112 1 001
5. Wrastawa Ridwan, ST., MT
NIP.19790205 200501 1 002

Gorontalo, Rabu 8 Juli 2020

Telah diterima sebagai salah satu persyaratan
untuk memperoleh gelar Sarjana Teknik (ST)



Dekan Fakultas Teknik
Universitas Negeri Gorontalo

Dr. Sardi Salim
NIP.19680705 199702 1 001

PERSETUJUAN PEMBIMBING

SKRIPSI

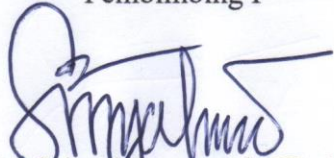
Rancangan Catu Daya Cadangan SRAM (Static Random Access Memory) Pada
Z80 Trainer

Oleh:

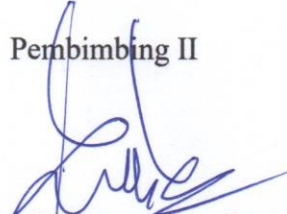
Hidayat Tahir
NIM.521413044

Telah diperiksa dan disetujui untuk oleh pembimbing

Pembimbing I


Syahrir Abdussamad, ST., MT
NIP.19750624 200501 1 003

Pembimbing II


Iskandar Zulkarnain Nasibu, S.Pd., M.Eng
NIP.19701104 200112 1 001

Mengetahui,
Ketua Jurusan Teknik Elektro


Yasin Mohamad, ST., MT
NIP.19710222 200112 1 001

RANCANGAN CATU DAYA CADANGAN SRAM (STATIC RANDOM ACCESS MEMORY) PADA Z80 TRAINER

Hidayat Tahir

Abstrak

Z80 Trainer banyak digunakan dalam praktikum maupun alat bantu peraga terkait pembelajaran mikroprosesor. Dalam penggunaannya, *Z80 Trainer* menggunakan *keypad* untuk menulis kode sumber secara manual dan selanjutnya disimpan dalam memori secara *volatile*. Masalah yang sering terjadi adalah putusnya pasokan daya ke *Z80 Trainer* sehingga program yang telah ditulis tidak tersimpan dan harus ditulis kembali. Penelitian ini ditujukan untuk merancang dan mengontrol catu daya cadangan untuk *SRAM* di *Z80 Trainer* menggunakan baterai, yang mampu mendeteksi terputusnya suplai daya dari tegangan listrik PLN untuk memastikan bahwa data dan program yang telah ditulis dan tersimpan di memori tidak terhapus. Metode yang digunakan dalam penelitian ini adalah metode perancangan alat pada laboratorium. Penelitian ini telah menghasilkan alat yang dapat mengontrol catu daya cadangan berasal dari baterai melalui rangkaian *switching control unit* untuk *SRAM* di *Z80 Trainer*. Hasil pengujian unjuk kerja mendapati bahwa alat yang telah dibuat dapat mengontrol atau mengatur catu daya di *SRAM* disaat catu daya utama PLN terputus selama kurang lebih 26 hari dengan memakai baterai *Li-Ion* tipe 18650 berkapasitas 3000 *mAh*. Pengujian lama waktu penggunaan alat pada kondisi normal menggunakan catu daya cadangan mendapatkan hasil sekitar ± 20 jam.

Kata kunci: *Z80 Trainer*, *SRAM*, Catu Daya Cadangan

BACKUP POWER SUPPLY DESIGN FOR SRAM (STATIC RANDOM ACCESS MEMORY) IN Z80 TRAINER

Hidayat Tahir

Abstract

Z80 Trainer is mostly used in practice or as teaching aids for microprocessor learning. In its application, Z80 Trainer uses a keypad to write the source code manually, which then is saved in memory volatile. The problem often occurred is the disconnection of power supply to Z80 Trainer; hence the program that has been written is not saved and must be re-written. This research aims at designing and controlling the backup power supply for SRAM in Z80 Trainer using a battery which is able to detect the disconnection of power supply from PLN Voltage to ensure that the data and the program that have been written and saved in memory are not deleted. It uses a method of designing tools at Laboratory. This research has produced a tool that can control the backup power supply made of battery through the switching control unit for SRAM in Z80 Trainer. The test result for the work shows that the tool can control or manage the power supply in SRAM when the main power supply of PLN has been disconnected for about 26 days using 3000 mAh, 1.8650 Li-Ion Battery. It also reveals that the duration of the use of tools in normal conditions using the backup power supply is ± 20 hours.

Keywords: Z80 Trainer, SRAM, Backup Power Supply

