

Recenzní posudek diplomové práce

Autor: Tomáš Buriánek

Vektorové řízení synchronního motoru s permanentními magnety s vývojovou deskou PYNQ-Z2

Předložená diplomová práce se věnuje vektorovému řízení synchronního stroje na platformě PYNQ-Z2, vývojové desce, která kombinuje programovatelné hradlové pole (FPGA) a mikroprocesor. Pro tvorbu řídicího algoritmu využil autor kompilaci programu ze Simulinku. Uživatelské rozhraní pak vytvářel v jazyku Python.

V práci student nejprve shrnuje základní principy FPGA a jejich vnitřní strukturu. Je zejména vyzdvížena schopnost paralelní práce jednotlivých částí, což je výhodné např. právě pro vektorové řízení, protože některé části algoritmu mohou běžet současně. Část textu je věnována metodám programování FPGA, s ukázkami kódu v jazyce Verilog. Protože student použil FPGA Xilinx, resp. dnes již AMD, používal vývojové prostředí Vivaldo, kterému se také v textu stručně věnuje.

V dalších kapitolách se student věnuje popisu principu vektorového řízení a modulace napětí. Student použil vývojovou desku PYNQ-Z2, s obvodem XC7Z020-1CLG400C. Jako výkonový modul pro připojení motoru pak modul BOOSTXL-DRV8305EVM a motor Technic M-2310P-LN-04K.

Řídicí algoritmus realizoval v Simulinku, s tím, že některé části jsou dělány „klasickými“ bloky, některé části jako stavové diagramy (např. ovládání A/D převodníku). Je implementováno měření napětí, fázových proudů a polohy rotoru. Pulzně šířková modulace (PWM) je generována symetrickým čítačem ze základního hodinového signálu.

Jedna část programu běží v FPGA, druhá v procesoru. Vzájemnou komunikaci realizoval student jako frontu přes přímý přístup do paměti (DMA). Realizace je v práci podrobně popsána.

Uživatelské rozhraní běží na PC, tj. nikoliv na desce PYNQ, komunikace je přes vlastní protokol po Ethernet TCP. Grafické uživatelské rozhraní umožňuje nastavovat parametry, číst data a celou aplikaci ovládat.

V kapitole 9 je ukázáno otestování celé aplikace, na jedné přechodové charakteristice proudu a rychlosti. Zde bych očekával více ukázaných výsledků, např. reakci na poruchu. V poslední kapitole je uveden popis použitých verzí SW a způsob živení, které vypadá netriviální. Uvedený postup jsem nezkoušel, ale nevím, jestli bych podle něj byl schopen bez pomoci autora vše zprovoznit.

Cíle práce považuji za splněné. Autor se musel seznámit s celou řadou oblastí. Od programování FPGA a jejich specifika, přes vytvoření řídicího algoritmu, generování a přenesení kódu až po experimenty. Práce je logicky popsaná a má požadovanou strukturu. Práci považuji za velice zdařilou.

Otázky:

- 1) Některá schémata, např. obr. 7.11 jsou velice nepřehledná. Pochybuji, že by se v nich někdo jiný kromě autora dokázal dobře orientovat. Umožňuje prostředí Vivaldo pojmenování vodičů a symbolické odkazy názvem vodiče? Pokud ano, proč jste je nepoužil? Schéma by mohlo být přehlednější.
- 2) Deska PYNQ-Z2 má i HDMI výstup. Bylo by možné připojit přímo monitor (a klávesnici) přímo k desce a spustit uživatelské rozhraní přímo na ní?
- 3) Proč do GUI přenášíte také regulační odchylku z regulátoru rychlosti?
- 4) Zmiňujete, že funkce sinus a cosinus jste realizovat formou tabulky. Uvažoval jste i jiné možnosti, např. algoritmus CORDIC?
- 5) S jakým krokem je možné nastavovat konstanty regulátorů?

Práci doporučuji k obhajobě a navrhuji hodnocení diplomové práce známkou:

„A - výborně“

Doc. Ing. Martin Novák Ph.D
Ústav přístrojové a řídicí techniky
Fakulta strojní ČVUT v Praze