

COMPARING COMPUTATIONAL DEMANDINGNESS OF RMS ALGORITHMS IMPLEMENTED IN ADSP2181

Miroslav KRUPA, Master Degree Programme (2)
Dept. of Control and Measurement, FEEC, BUT
E-mail: xkrupa03@stud.feec.vutbr.cz

Supervised by: Dr. Petr Beneš

ABSTRACT

In this work author compares different algorithms of computing RMS (Root Mean Square) in time domain. Algorithms is comparing in view of computing time and memory requirements. Algorithms are written in C language and are designed for DSP (Digital Signal Processor) of Analog Devices ADSP2181. Comparing is realized in Simulator and Debugger of VisualDSP++ software.

1 ÚVOD

Tato práce se zabývá číslicovým zpracováním signálů pomocí signálového procesoru ADSP2181, konkrétně se jedná o porovnání algoritmů výpočtu RMS signálu z hlediska paměťových nároků, náročnosti a délky výpočtu.

DSP nacházejí díky svým nesporným výpočetním výkonům, uplatnění nejenom v telekomunikační a audiovizuální technice, ale i technice měřicí a řídicí.

Cílem této práce je ověřit možnosti využití signálového procesoru při měření např. vibrací a výpočtu efektivní hodnoty. Nejedná se o řešení konkrétních požadavků výpočtu pro praktické měření, ale spíše o obecný pohled na problematiku výpočtů a implementace algoritmů do DSP.

2 ROZBOR

2.1 TEORETICKÝ ROZBOR

Efektivní hodnota signálu je důležitou charakteristickou hodnotou časového průběhu napětí. Uvedme si možnosti číslicového zpracování signálu pro výpočet efektivní hodnoty RMS. Diskrétní signál je možno zpracovávat v časové oblasti a v oblasti frekvenční (při zpracování se používá koeficientů FFT reprezentující jednotlivé frekvenční složky).

V našem případě nás zajímá výpočet efektivní hodnoty v časové oblasti, který je matematicky definována následovně:

$$U_{RMS} = \sqrt{\frac{1}{T} \int_0^T u^2(t) dt}, \quad (1)$$

kde T je perioda, má-li napětí $u(t)$ periodický charakter. Převědeme-li do diskrétního tvaru je integrace nahrazena sumací:

$$U_{RMS} = \sqrt{\frac{\sum_{k=1}^N u_k^2(kT)}{N}}, \quad (2)$$

kde N je počet vzorků, u_k je hodnota k .tého vzorku, T je perioda vzorkování. Výpočet lze tedy realizovat jako posloupnost operací druhé mocniny, sumace a následné druhé odmocniny.

V případě měření např. vibrací, které mají nulovou střední hodnotu tj. nemají stejnosměrnou složku, se může uplatnit stejnosměrná fluktuace předřazených odporů v měřicím obvodu. Tj. při výpočtu podle (2) by jsme počítali nejen samotnou efektivní hodnotu vibrací, ale ve výpočtu by byla zahrnuta i složka stejnosměrná, která zkresluje výsledek a nemá zde opodstatnění. Pokud chceme potlačit tuto stejnosměrnou složku užijeme následující vzorec:

$$U_{RMS} = \sqrt{\frac{1}{N} \sum_{k=1}^N (u_k(kT) - U_{stř})^2}, \quad (3)$$

kde $U_{stř}$ je definovaná následovně

$$U_{stř} = \frac{1}{N} \sum_{k=1}^N u_k(kT) \quad (4)$$

Vzorec (3) byl upraven dohlížitelem této práce tj. Ing Petrem Benešem, PhD. do vhodnějšího tvaru pro implementaci v signálovém procesoru. V tomto případě není nutné uchovávaní vzorků, což snižuje nároky na výpočet:

$$U_{RMS} = \sqrt{\frac{1}{N} \left(\sum_{i=1}^N u_k^2(kT) - \frac{1}{N} \left(\sum_{i=1}^N u_k(kT) \right)^2 \right)} \quad (5)$$

2.2 SROVNÁNÍ ČASOVÉ NÁROČNOSTI ALGORITMŮ RMS

Časová náročnost resp. délka výpočtu jednotlivých algoritmů je odvozena od doby instrukčního cyklu (v našem případě má instrukční cyklus délku 30ns) a od počtu těchto instrukcí. Byl zjištěn následující počet instrukcí pro $N=1024$:

Algoritmus podle vzorce 2:

$28(\text{počáteční nastavení proměnných}) + N \cdot 23(\text{sumace a výpočet kvadrátů}) + 530(\text{podíl } N) + 950(\text{výpočet odmocniny}) = 25060$.

Doba trvání výpočtu tedy je

$\text{počet instrukcí} \cdot \text{doba instrukčního cyklu} = 25060 \cdot 30.0e-9 = \underline{0,75ms}$

Algoritmus podle vzorce 3:

$30(\text{počáteční nastavení proměnných}) + N \cdot 51(\text{výpočet střední hodnoty}) + N \cdot 70(\text{sumace a výpočet kvadrátů}) + 530(\text{podíl } N) + 950(\text{výpočet odmocniny}) = 125414$.

Doba trvání výpočtu tedy je

$\text{počet instrukcí} \cdot \text{doba instrukčního cyklu} = 125414 \cdot 30.0e-9 = \underline{3,81ms}$

Algoritmus podle vzorce 5:

$31(\text{počáteční nastavení proměnných}) + N.36(\text{výpočet střední hodnoty a sumace}) + 529(\text{výpočet kvadrátů střední hodnoty}) + 950(\text{výpočet odmocniny}) = 38374.$

Doba trvání výpočtu tedy je :

$$\text{počet instrukcí} \cdot \text{doba instrukčního cyklu} = 38374 \cdot 30 \cdot 10^{-9} = \underline{1,17\text{ms}}$$

Z výše uvedeného je patrné, že výpočet podle vzorce (5) je více než 3x rychlejší než podle vzorce (3). Uplatňuje se zde prodleva při výpočtu střední hodnoty.

Všechny výše uvedené časy resp. počty instrukcí jsou zjištěny s použitím výpočtů v plovoucí čárce, které překladač vyššího programovací jazyka C podporuje. Díky tomuto jsou výpočty sice pomalejší, ale daleko přesnější (nedochází zde k saturacím a podtečení).

2.3 SROVNÁNÍ PAMĚŤOVÉ NÁROČNOSTI ALGORITMŮ RMS

Algoritmy lze také srovnat podle paměťových nároků. Paměť se skládá ze dvou částí tj. z programové (obsahující instrukce pro vykonávání programu) o velikosti 16kWords (tj. $16 \cdot 1024 \cdot 24$ bitů) a datové paměti (obsahující data užitá v programu) o velikosti 16kWords (tj. $16 \cdot 1024 \cdot 16$ bitů) . Byly zjištěny tyto následující požadavky na paměť pro jednotlivé algoritmy :

Algoritmus podle vzorce (2):

Programová paměť : $4209 \text{ Bytů} = 1,37 \text{ kWords}$

Datová paměť: $170 + 2 \cdot 2 = 174 \text{ Bytů} = 0,08 \text{ kWords}$

Algoritmus podle vzorce (3):

Programová paměť : $4764 \text{ Bytů} = 1,58 \text{ kWords}$

Datová paměť: $170 + 2 \cdot 1024 = 2218 \text{ Bytů} = 1,08 \text{ kWords}$

Algoritmus podle vzorce (5):

Programová paměť : $4410 \text{ Bytů} = 1,43 \text{ kWords}$

Datová paměť: $228 + 2 \cdot 2 = 232 \text{ Bytů} = 0,11 \text{ kWords}$

3 ZÁVĚR

Těžištěm a potažmo cílem této práce mělo být zejména seznámení se s problematikou signálových procesorů a také s problematikou realizace výpočtu algoritmu RMS. Při simulacích byl použit obecný sinusový signál. Nepřesnost měření resp. výpočtu vzhledem k užití plovoucí řádové čárky je pro všechny algoritmy zanedbatelná. Bylo zjištěno, že výpočet podle vzorce (5) je mnohem výhodnější než podle vzorce (3) z hlediska časové i paměťové náročnosti (není potřeba uchovávat všechny vzorky, stačí jen aktuální vzorek). Tato práce je stručné shrnutí Semestrální práce 2 .

LITERATURA

- [1] Jílek, P.: Využití ADSP21xx pro úpravu signálu, Diplomová práce VUT Brno, 2001 + součástí CD-ROM
- [2] Analog Devices, Inc.: ADSP 21xx Family User's Manual, 3.vydání Analog Devices Inc., 1995, ISBN 82-000780-03.
- [3] Analog Devices, Inc.: Mixed-Signal and DSP Design Techniques, Analog Devices Inc., 2000, ISBN-0-916550-23-0.