



GENERATOR TRAJEKTORIJE ZA VEKTORSKI KONTROLISANI ASINHRONI POGON PRIMENOM DSP-a

Milan Živić

Tehnička škola, Kikinda, Srbija i Crna Gora, zires@eunet.yu

Sadržaj: Detaljno je opisan algoritam za generisanje trajektorije kretanja vratila vektorski regulisanog asinhronog motora. Posebno su obrađeni pozicioni i brzinski mod rada. Izvršeno je modelovanje, a zatim i praktična implementacija generatora trajektorije u asembleru. Iskorišćen je Digital Signal Procesor TMS320F240 koji se standardno upotrebljava u Motor Control aplikacijama.

Ključne reči: DSP, generator trajektorije.

1. UVOD

Kvalitetan generator trajektorije kretanja je neophodan za precizno pozicioniranje vratila motora bez premašaja i neželjenih oscilacija. Zbog toga se njemu mora posvetiti posebna pažnja. Svako kretanje ima svoje parametre (granična brzina, ubrzanje, početna i krajnja pozicija) koji se mogu nezavisno podešavati. Generisanje trajektorije veoma zavisi i od mehaničkog dela sistema na koji se vratilo motora povezuje (korak zavojnog vretena, prenosni odnos, itd.).

Pomenuti generator trajektorije je iskorišćen za zadavanje referentne pozicije kod razvojnog modela vektorskog regulatora i pokazao je odlične performanse. Algoritam je rešio brojne probleme praktične prirode na koje je autor nailazio u praksi sa asinhronim pogonima.

2. GENERISANJE TRAJEKTORIJE KRETANJA

Generator trapezne trajektorije brzine računa zahtevanu poziciju motora u vremenu.

U pozicionom modu rada definišu se: *ubrzanje*, *maksimalna brzina* i *krajnja pozicija*. DSP koristi ove informacije da bi ostvario pokret sa specificiranim ubrzanjem, dok se ne dostigne max brzina ili, dok ne počne da usporava da bi se zaustavio na navedenoj krajnjoj poziciji. Nagibi usporenja i zaleta su jednaki. Parametri trajektorije mogu se menjati samo nakon završenog kretanja.

U brzinskom modu rada motor se zaleće do specificirane brzine sa navedenim ubrzanjem i održava tu brzinu do stop komande. Brzina se održava konstantnim kretanjem unapred zahtevane pozicije.

Željena i stvarna pozicija su označene 16-to bitne vrednosti. Ubrzanje je 16-to bitna, a max brzina 32-bitna

pozitivna varijabla koja ima 16-to bitni razlomljeni deo. Slede izrazi za kalkulaciju željene brzine $V(k)$ [imp/sempl] i pozicije $\theta(k)$ [imp].

$$V(k) = V(k-1) + a \cdot K_a \cdot T_{poz} \cdot h(x) - a \cdot K_a \cdot T_{poz} \cdot h(y) \quad (1)$$

$$\theta(k) = \theta(k-1) + \frac{1}{65536} \cdot V(k) \cdot T_{poz} \quad (2)$$

gde su:

- a – konstantno ubrzanje u [mm/s²],
- $a \cdot K_a$ – ubrzanje u [imp/sempl²] sa kojim kalkuliše DSP,
- $h(x)$ – logička funkcija koja je jednaka jedinici u toku zaleta, dok je u svim ostalim slučajevima nula,
- $h(y)$ – logička funkcija koja je jednaka jedinici u toku kočenja, dok je u svim ostalim situacijama nula.

Prilikom zaletanja motora trenutna brzina $V(k)$ se dobija stalnim dodavanjem zadatog ubrzanja prethodnom sadržaju brzinskog registra ($h(x)=1$), sve dok se ne dostigne postavljena max brzina (V_{lim}). Naravno, u pitanju je *diferencni* račun koji se ponavlja sa periodom T_{poz} (sempl). Prilikom kočenja motora imamo obrnutu situaciju, jer je tada $h(y)=1$. Trenutna referentna pozicija $\theta(k)$ dobija se jednostavnim sabiranjem prethodne pozicije sa *celobrojnim* delom brzine (otuda deljenje sa $2^{16}=65536$ u izrazu (2)). Važno je napomenuti da se u registar željene pozicije unose samo celobrojne vrednosti $\theta(k)$ [imp], dok se pri kalkulaciji koristi i razlomljeni broj impulsa. Sve će biti mnogo jasnije na konkretnom primeru, ali je pre njega potrebno definisati i odrediti konstante ubrzanja (K_a) i brzine (K_v). Ove konstante se koriste za prilagođenje između jedinica ubrzanja i brzine kojima se služi korisnik, i formata zapisa istih unutar DSP-a. Željeno ubrzanje se zadaje softverski u [mm/s²], dok se max brzina programira u [μm/s]. Ovo je razumljivo s obzirom da se na taj način definiše linearno kretanje. DSP zahteva da ubrzanje bude uneto u [imp/sempl²], a brzina u [imp/sempl]. Iz tog razloga moraju se odrediti odgovarajući faktori skaliranja (K_a i K_v):

$$a \left[\frac{\text{imp}}{\text{semp}^2} \right] = K_a \cdot a \left[\frac{\text{mm}}{\text{s}^2} \right] \quad (3)$$

$$V \left[\frac{\text{imp}}{\text{semp}} \right] = K_v \cdot V \left[\frac{\mu\text{m}}{\text{s}} \right] \quad (4)$$

Ti faktori obezbediće da DSP generiše trajektoriju koja će fizički odgovarati programiranim parametrima:

$$K_v = 65536 \cdot T_{\text{poz}} \cdot \frac{4E}{K} \cdot \left(\frac{z_2}{z_1} \right) \quad (5)$$

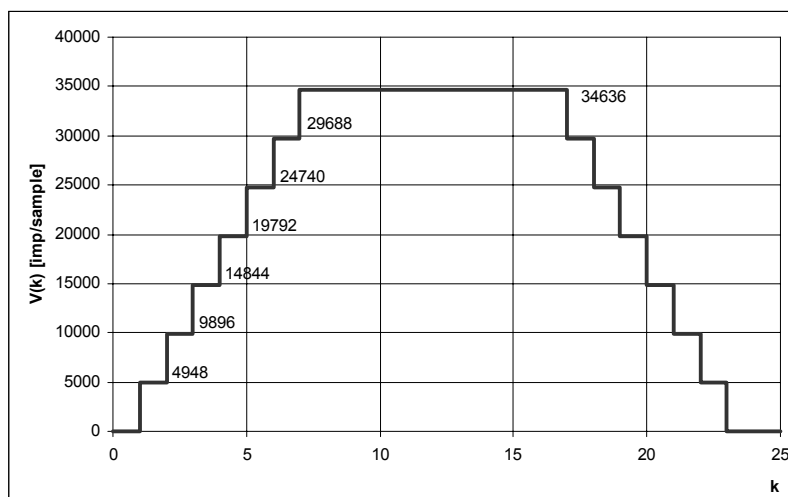
$$K_a = 65536 \cdot T_{\text{poz}}^2 \cdot 10^3 \cdot \frac{4E}{K} \cdot \left(\frac{z_2}{z_1} \right) \quad (6)$$

gde su:

- E – broj impulsa enkodera [imp/obr],
- K – konstanta zavojnog vretena [$\mu\text{m}/\text{obr}$],
- (z_2/z_1) – prenosni odnos zupčanika na vretenu i na motoru.

Zamenom vrednosti (za konkretan mehanički sistem):

$E=5000$ imp/obr, $K=5000$ $\mu\text{m}/\text{obr}$, $z_2=2 \cdot z_1$ i $T_{\text{poz}}=307.2$ $\mu\text{s}/\text{semp}$, dobija se: $K_v=161.06$ (imp·s)/(semp· μm) i $K_a=49.478$ (imp·s²)/(semp²·mm).



Sl.1. Generisanje referentne brzine (primer)

Izračunate konstante se zaokružuju ($K_v=161$, $K_a=49.48$) i unose se u asemblerski program koji upravlja radom glavnog procesora.

Osim pomenutih faktora za skaliranje brzine i ubrzanja, neophodno je odrediti i *prenosni odnos* koji povezuje zadatu poziciju u [μm] sa pozicijom u [imp]. Ponovo se mora razmotriti mehanički deo sistema:

- koristi se zavojno vreteno od 5000 $\mu\text{m}/\text{obr}$,
- na vretenu se nalazi zupčanik sa z_2 zubaca, dok je na vratilu motora zupčanik sa z_1 zubaca,
- $z_2=2 \cdot z_1$, odnosno jednom obrtaju motora odgovara pola obrtaja vretena,
- enkoder na vratilu motora ima 5000 imp/obr,
- jednom obrtaju motora odgovara $4 \cdot E=20000$ impulsa odbrojanih od strane DSP-a,
- za jedan obrtaj motora dobija se linearni pomeraj ose od $(z_1/z_2) \cdot 5000 \mu\text{m}=2500 \mu\text{m}$.

Na osnovu svega navedenog, može se zaključiti da fizičkom pomeraju od 2500 μm odgovara 20000 impulsa (rastućih i opadajućih ivica sa enkodera). Odnosno, traženi *prenosni odnos* je 1:8 (1 $\mu\text{m}=8$ imp). Ovo praktično znači da je željenu poziciju [μm] prilikom upisa u DSP potrebno pomnožiti sa 8. Tada će postojati usaglašenost između realnog fizičkog pomeraja i onoga što imamo programirano. Sada postoje svi relevantni podaci za analizu zadavanja jednog konkretnog kretanja u pozicionom modu rada.

Primer: generisanje trapezne trajektorije brzine i referentne pozicije sa sledećim parametrima:

- $a = 100 \text{ mm}/\text{s}^2$,
- $V_{\text{lim}} = 200 \mu\text{m}/\text{s}$,
- put = 1 μm .

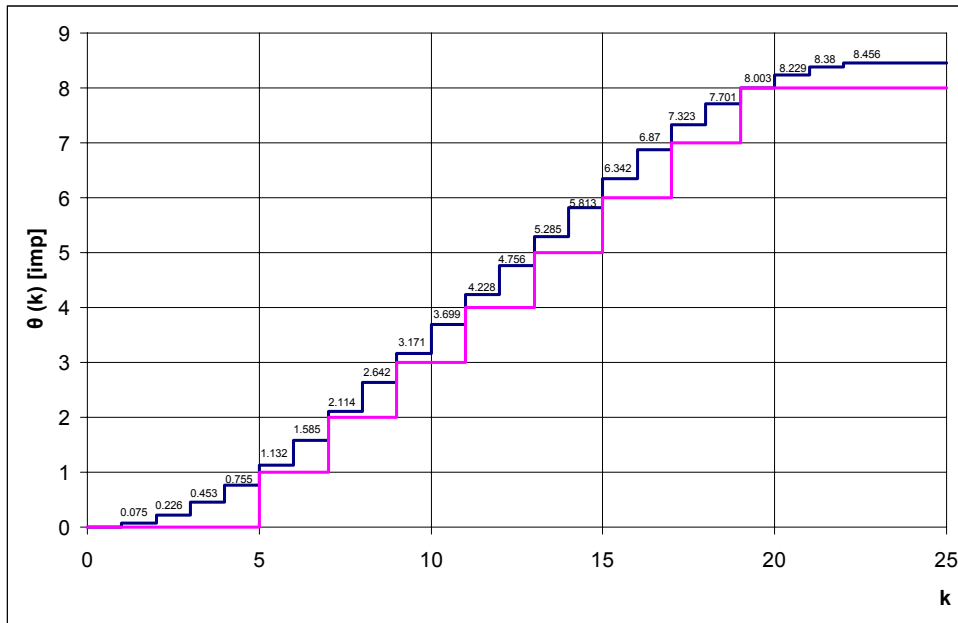
Na osnovu određenih konstanti $K_a=49.48$ i $K_v=161$, kao i na osnovu poznavanja prenosnog odnosa (1:8), lako se dolazi do vrednosti koje se upisuju u DSP (izrazi (3) i (4)):

- a [imp/semp²] = 4948,
- V_{lim} [imp/semp] = 32200,
- put [imp] = 8.

Sl. 1. ilustruje sadržaj brzinskog registra koji se generiše po *diferencnoj* jednačini (1). Vreme zaleta jednako je vremenu kočenja i iznosi $7 \cdot T_{\text{poz}}=2.15$ ms. Evidentno je vrlo malo odstupanje od planiranih (V_{lim}/a)=2 ms. Stvarna max brzina iznosi $(34636/161)$ $\mu\text{m}/\text{s}=215 \mu\text{m}/\text{s}$. Ukupno kretanje traje 24 periode odabiranja ($24 \cdot T_{\text{poz}} \approx 7.37$ ms).

Sl. 2. predstavlja referentnu poziciju izračunatu po izrazu (2), gde se uzima u obzir uticaj razlomljenog dela brzine $V(k)$. Drugom krivom naznačen je sadržaj registra željene pozicije koji uzima samo celobrojne vrednosti impulsa. Za $k=0, \dots, 7$ (zalet) pozicija $\theta(k)$ menja se po paraboličnom zakonu. Od $k=8$ do $k=17$ (V_{lim}) imamo linearnu promenu pozicije sa konstantnim priraštajem od $(34636/65536)$ imp/semp ≈ 0.5285 imp/semp. Prilikom kočenja ($k=18, \dots, 24$) ciljna pozicija se zauzima ponovo po paraboličnoj krivi. Uzimanjem samo celobrojnih vrednosti $\theta(k)$ dolazi se do referentne pozicije.

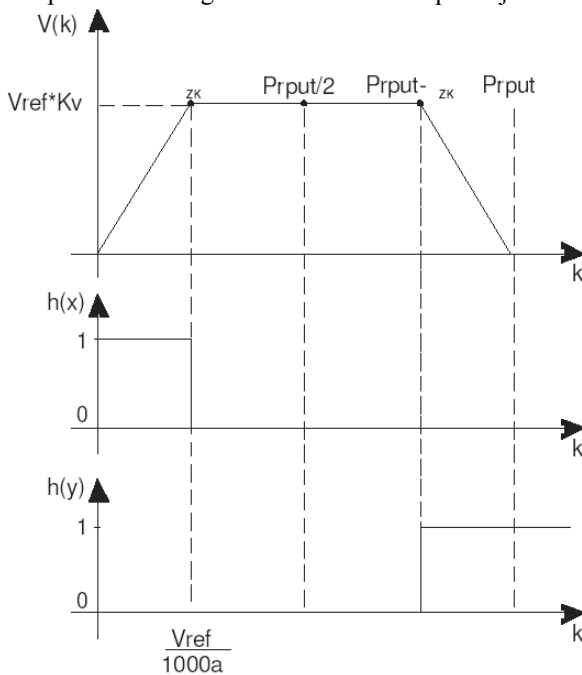
U pozicionom PD algoritmu učestvuju samo željena i stvarna pozicija. Izlazna, tj. upravljačka promenljiva dobija se težnjom kompenzatora da ostvari programirani željeni tempo (nagib) porasta referentne pozicije. Ta upravljačka promenljiva u stvarnosti igra ulogu trajektorije brzine i vodi se kao *referenca brzine* na vektorski regulator.



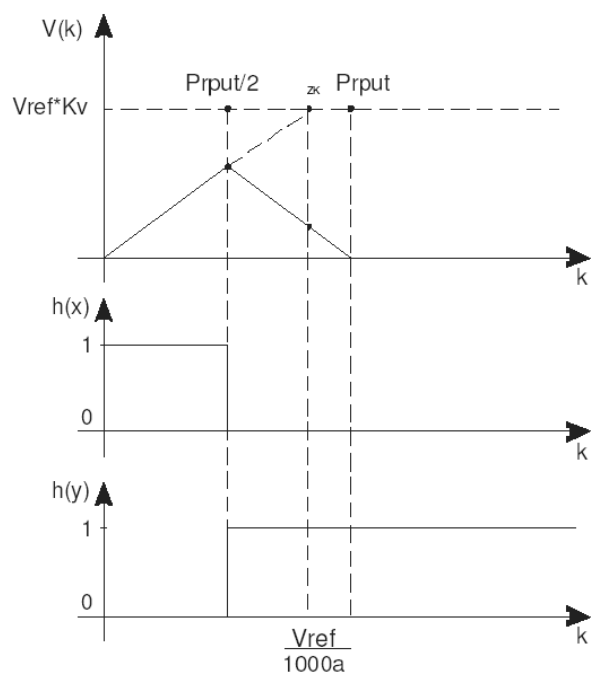
Sl. 2. Generisanje referentne pozicije (primer)

Ako se npr. javi premašaj brzine motora, stvarna pozicija dobija veći nagib od željene ($e(k) < 0$) i pozicioni PD smanjuje svoj izlaz kako bi se smanjila i stvarna brzina obrtanja. Ravnotežu (konstantan *set*) imamo kada se uspostave isti nagibi zadate i aktuelne pozicije.

Brzinski mod rada sličan je pozicionom, samo što kod njega nema fiksirane krajnje pozicije, nego se referentna pozicija stalno povećava sa nagibom koji odgovara trenutnoj željenoj brzini.



Sl. 3. Trapezna trajektorija brzine



Sl. 4. Trougona trajektorija brzine

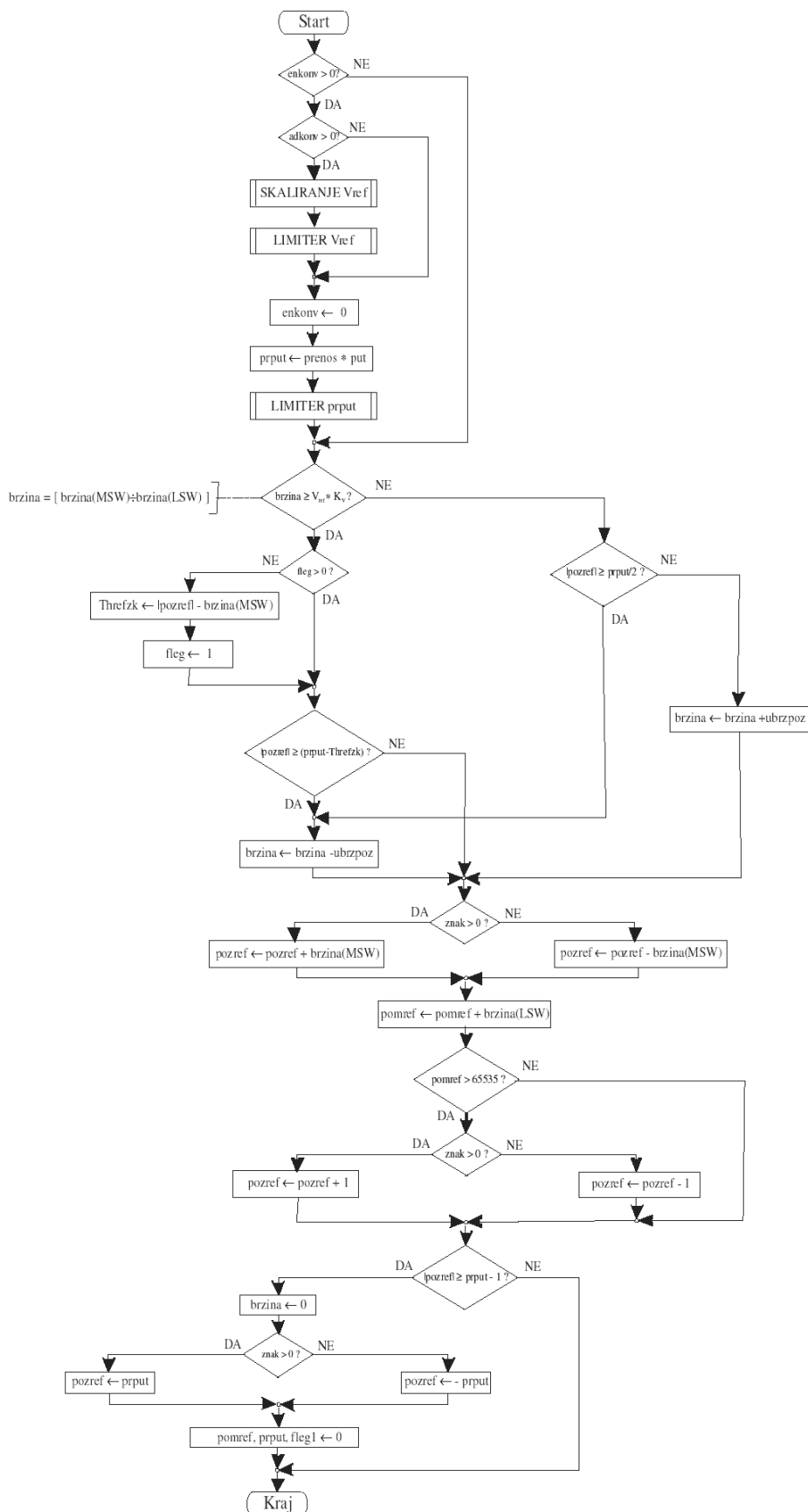
Ako postoje smetnje u obrtanju (motor teško ide ili se zaglavio) ili kada mehanički deo sistema ne može da prati zahtevanu dinamiku kretanja, referentna pozicija će stalno rasti uzrokujući veoma veliku pozicionu grešku. Prekomerna poziciona greška mora se detektovati, jer predstavlja ozbiljan sistemski problem. U tu svrhu koristi se parametar GRESKA, kojim se definiše maksimalno dozvoljeni "prozor" apsolutne pozicione greške na kraju kretanja. Ovaj parametar je pozitivna 16-to bitna vrednost i zadaje se u $[\mu\text{m}]$. Teži se što manjem pragu greške, jer se tada ostvaruje kvalitetna kontrola pozicione konture regulacije. Ako poziciona greška izade iz

specificiranog prozora, automatski se stopira elektromotorni pogon. Maksimalna vrednost koja se može upisati iznosi $4095 \mu\text{m}$ (zbog množenja sa 8 prilikom upisa u DSP).

DSP može generisati po obliku dve različite trajektorije brzine:

- trapeznu (kada se dostiže $V_{lim} = V_{ref} K_V$), i
- trougaonu (kada se V_{lim} ne dostiže pre završetka kretanja).

Na slikama 3. i 4. prikazana su oba slučaja, pri čemu su date i sve karakteristične tačke putanje, kao i izgled logičkih funkcija za zalet i kočenje ($h(x)$ i $h(y)$).



Sl. 5. Algoritam generatora trajektorije za pozicioni mod rada

3. ALGORITAM GENERATORA TRAJEKTORIJE KRETANJA

Na slici 5. predstavljen je algoritam generatora trajektorije za *pozicioni mod* rada. Zbog nedostatka

prostora *brzinski mod* nije obrađen. Takođe je izostao i algoritam *glavnog programa* koji kontroliše uslove za start regulatora, ispituje mod rada, generiše sve periode odabiranja, ispituje da li je trajektorija zaista fizički pređena, skanira tastaturu i ulaze, postavlja izlaze, flegove i resetuje odgovarajuće promenljive.

Sledi spisak svih korišćenih varijabli u algoritmu sa slike 5.:

- *enkonv* – fleg varijabla koja ne dozvoljava promenu maksimalne brzine i krajnje pozicije za vreme kretanja,
- *adkonv* – fleg varijabla koja selektuje način zadavanja maksimalne brzine kretanja (programski ili preko potencijometra),
- *prput* – neoznačena varijabla koja predstavlja željenu poziciju u *[imp]*,
- *prenos* – neoznačena varijabla koja definiše prenosni odnos (npr. $1 \mu m = 8 imp \Rightarrow prenos = 8$),
- *brzina* – neoznačena 32-bitna varijabla koja predstavlja registar trenutne referentne brzine obrtanja $V(k)$,
- *brzina (MSW)* – viša reč brzine $V(k)$,
- *brzina (LSW)* – niža reč brzine $V(k)$ ili razlomljeni deo brzine,
- *Threfzk* – neoznačena 16-to bitna varijabla koja predstavlja vrednost referentne pozicije (θ_{zk} sa slike 3.) u trenutku kada postaje $V(k) \geq V_{ref} \cdot K_V$, odnosno na kraju zaleta,
- *pozref* – označena 16-to bitna varijabla referentne pozicije u *[imp]*,
- *ubrpoz* – ubrzanje za pozicioni mod rada $a[imp/sempl^2]$,
- *znak* – fleg varijabla smera obrtanja motora,
- *pomref* – pomoćna neoznačena varijabla koja uzima u obzir razlomljeni deo brzine prilikom računanja referentne pozicije,
- *fleg* – fleg varijabla koja obezbeđuje da se θ_{zk} određuje samo jednom po trajektoriji kretanja, i
- *fleg1* – fleg varijabla za završetak referentne trajektorije kretanja.

Kroz algoritam generatora trajektorije se prolazi svakih $T_{poz} = 307.2 \mu s$, sve dok traje kretanje. Na početku je *enkonv* = 1, što omogućava definisanje granične brzine V_{ref} i krajnje pozicije *prput* (ubrzanje *ubrpoz* zadaje se isključivo programski). U toku kretanja parametri se ne mogu menjati, jer je *enkonv* = 0. Maksimalna brzina V_{ref} se može zadavati programski (*adkonv* = 0) ili preko potencijometra (*adkonv* = 1). Ukoliko se zadaje preko potencijometra, obavlja se skaliranje i limitiranje naponskog signala nakon A/D konverzije, kako bi se mogao podesiti željeni opseg brzina.

Sledi upitnik završenog zaleta ($brzina \geq V_{ref} \cdot K_V$), kojim se ispituje da li je dostignuta granična brzina. Ukoliko nije dostignuta, ispituje se da li je pređena polovina željenog puta, kako bi se mogla generisati trougaona trajektorija (slika 4.):

- za pređenu polovinu puta sledi kočenje ($brzina \leftarrow brzina - ubrpoz$),
- za nepređenu polovinu puta sledi dalje zaletanje ($brzina \leftarrow brzina + ubrpoz$).

Ukoliko je granična brzina dostignuta, pamti se referentna pozicija koja odgovara trenutku završetka zaleta ($Threfzk = \theta_{zk}$), kako bi se u slučaju trapezne trajektorije mogao odrediti momenat kada treba preći na kočenje. Uslov za kočenje je $|pozref| \geq (prput - \theta_{zk})$, što se lepo vidi i sa slike 3.

Referentna pozicija *pozref* računa se na osnovu celobrojnog dela brzine (viša reč) korišćenjem izraza (2). Pošto je brzina uvek pozitivna, upotrebljava se varijabla *znak* da odredi smer kretanja. Razlomljeni deo brzine (niža reč) se uzima u obzir preko promenljive *pomref*. Na svakih 65536 dodaje se ili oduzima jedan impuls referentnoj poziciji. Na kraju algoritma nalazi se zaštitna petlja koja osigurava da na završetku trougaone trajektorije uvek bude $|pozref| = prput$ i $brzina = 0$. Takođe se resetuju sve relevantne varijable.

4. ZAKLJUČAK

Za realizaciju asinhronog pogona visokih performansi neophodno je, osim algoritma vektorske regulacije obezbediti niz drugih uslova:

- precizno merenje stvarne pozicije i brzine obrtanja motora,
- brzinski PI regulator sa adaptivnom promenom parametara u zavisnosti od trenutne brzine obrtanja,
- digitalni PD pozicioni kompenzator sa optimalno podešenim parametrima,
- optimalno izabrane periode odabiranja strujnih, brzinske i pozicione regulacione petlje, itd.

Međutim, bez kvalitetnog generatora reference trajektorije kretanja ne može se zamisliti precizno pozicioniranje bez premašaja i dodatnih oscilacija. Zbog toga je razvijen softverski generator koji omogućava linearni zalet i kočenje motora, odnosno parabolično startovanje i zauzimanje referentne pozicije. Opisani algoritam je primenjen kod razvojnog modela vektorskog regulatora za asinhroni motor i u praksi je pokazao odlične karakteristike.

5. LITERATURA

- [1] M. Živić, "Primena vektorski regulisanog asinhronog pogona za pozicioniranje ose na CNC mašini u Livnici Kikinda AD", Magistarski rad, FTN, Novi Sad, 2002.
- [2] TI Documentation, "TMS320C2xx Assembly Language Tool", March 1998.
- [3] TI Documentation, "TMS320C240/F240 DSP Controllers Data Sheet", May 1998.
- [4] Dokumentacija i katalozi proizvođača LM628, National Semiconductors.

TRAJECTORY GENERATOR FOR INDUCTION DRIVE WITH VECTOR CONTROL USING DSP

Abstract: Algorithm of motion trajectory generation for induction motor with vector control is described comprehensive. Position and speed mod are separately considered. Trajectory generation is modeled and then implemented in assembler. A standard Motor Control Digital Signal Processor TMS320F240 is used for practical implementation.

Key Words: DSP, trajectory generator.