

ОРГАНИЗАЦИЯ НА DMA-ОБМЕН ЗА ISA-БАЗИРАНИ A/D-КОНТРОЛЕРИ

Димитър Ст. Тянев

Резюме: *Дискутират се пределните възможности за въвеждане на данни чрез каналите за пряк достъп до паметта от ISA-базирани контролери, предназначени за синхронно и асинхронно аналого-цифрово преобразуване. Изясняват се изискванията на процеса на обмен, необходими за синтез. Възможните технически и програмни решения се основават на два от режимите за работа на DMA-контролера - режим на единичен обмен и режим на блочен обмен.*

Key words: DMA-контролер, A/D-контролер.

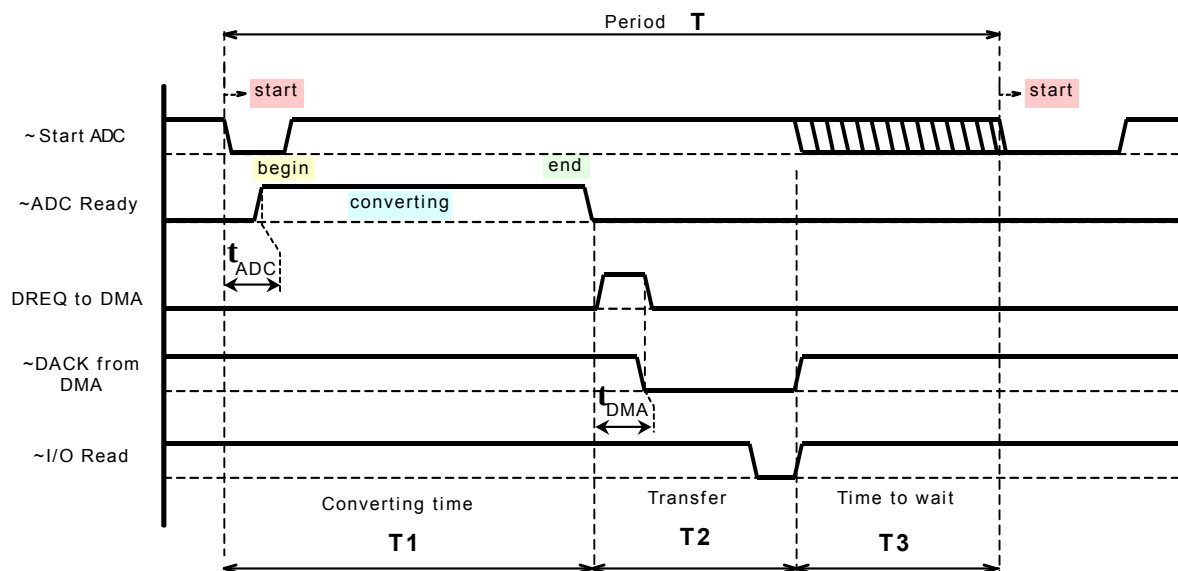
1. Пряк достъп до паметта

За изследване на един реален процес компютърната система се нуждае от неговата цифрова проекция. Възможността тя да бъде получена се осигурява от апаратно-програмни средства на една система за сбор на данни. Тези средства не са традиционни за масово разпространените компютърни конфигурации. Стремехът към простота в структурата и управлението на тази система водят към разработване на аналого-цифров контролер (АЦ-контролер), базиран върху системната компютърна шина, например според стандарта ISA. Лишаването на потребителския АЦ-контролер от собствена памет означава, че той ще използва наличната оперативна памет, достъпът до която ще бъде периодичен в циклическото повторение на операциите преобразуване и въвеждане. При тези условия и като се имат предвид високите изисквания спрямо стабилността на честотата за дискретизация на аналоговия сигнал, които се постигат чрез кварцова стабилизация, се стига до убеждението, че управлението на аналого-цифровото преобразуване и четенето на данни от АЦ-контролера е най-добре да се осъществява чрез сигналите на DMA-контролера. Обикновено компютърните платформи предлагат свободни за потребителя DMA-канал за 8-битов трансфер и DMA-канал за 16-битов трансфер. Подходящи за реализация на трансфера на данни от АЦ-контролера към оперативната памет са два от режимите за работа на DMA-канала - режим на единичен обмен и режим на блочен обмен.

А) Режим на единичен обмен

Когато DMA-каналът е програмиран за работа в този режим (битове b7,b6 в регистъра на режима имат стойностите b7=0, b6=1), прехвърлянето на всеки отделен отчет получен в АЦ-контролера към паметта на процесора, се осъществява по инициатива на контролера. Това той постига като подава към DMA-канала заявка за обслужване. В отговор DMA-контролерът иска от процесора правото да управлява системната шина и след като го получи осъществява трансфер до съответната клетка в оперативната памет на формирания в АЦ-контролера отчет. С това задълженията на DMA-контролера се изчерпват, той освобождава системната шина и остава в състояние на очакване. Процесите на преобразуване и трансфер на данни в този режим на работа на DMA-канала са илюстрирани на фигура 1.

От времедиаграмата се вижда, че темпът на аналого-цифровото преобразуване се задава от тактовия сигнал **Start ADC**. Всеки заден фронт на този сигнал стартира ново преобразуване при което отчетите на амплитудите следват с временна стъпка **T**. В рамките на периода **T** се наблюдават три етапа в развитието на процеса. Първият етап, с продължителност **T1**, принадлежи на аналого-цифровия преобразовател и се определя от неговите скоростни възможности. Интервалът **T1** започва от задния фронт на тактуващия сигнал **Start ADC** и завършва със задния фронт на сигнала **Ready**, който възниква в преобразователя, когато всички цифри на отчета са готови. Този временен интервал е постоянен и не може да се управлява, тъй като представлява техническа характеристика на самия преобразовател.



Фиг. 1

Вторият етап, с продължителност **T2**, принадлежи на DMA-контролера и се определя от продължителността на машинния цикъл върху системната шина. Началото му е в момента на възникване на сигнала **Ready** и завършва със записването на данните в паметта. В този временен интервал се изпълнява следната процедура: сигналът от преобразователя за готови данни **Ready** се използва от съответни схеми за формиране на заявка за трансфер **DREQ**, която се подава към съответния DMA-канал. Сигналят **DREQ** инициира в DMA-контролера действия за заемане на системната шина и прехвърляне на данни от АЦ-контролера към оперативната памет. След завършване на трансфера DMA-контролерът сменя потвърждаващия сигнал **DACK**, чийто преден фронт бележи края на този етап. Времето **T2** за трансфер на текущия отчет е малко повече от един машинен цикъл и не може да бъде променено.

Третият етап, с продължителност **T3**, по същество представлява време на очакване. В този временен участък както АЦ-контролерът така и DMA-каналът са в състояние на очакване. Първият очаква стартиращ фронт в сигнала **Start ADC**, а вторият - заявка **DREQ**.

Разнообразни технически решения, използващи описаната процедура за трансфер на данни чрез DMA-канал, работещ в режим на единичен обмен, са описани в [1], [2], [3], [4].

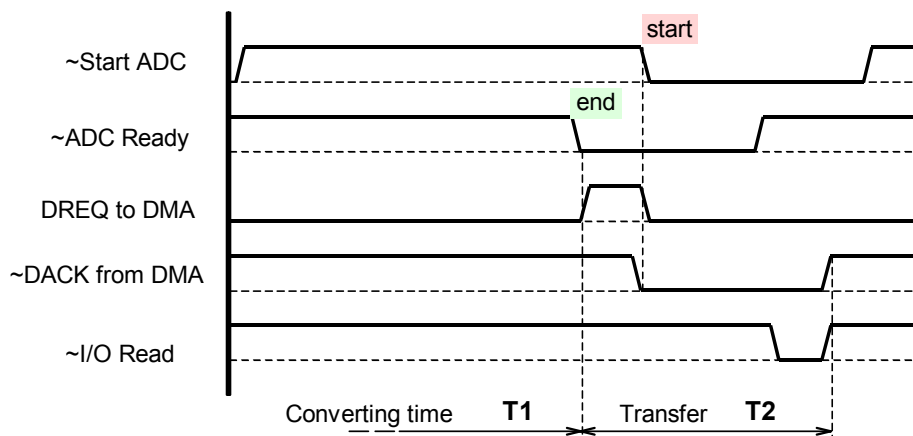
Интерес представляват скоростните възможности на една система за сбор на данни, използваща този метод за организация на трансфера, тъй като те определят нейното приложение. Ниските скорости на аналого-цифровото преобразуване се постигат чрез увеличаване на периода **T** на стартиращите импулси. Обикновено изменението на скоростта се реализира чрез програмно управляван делител на тактовата честота. Може да се каже, че постигането на ниски скорости на дискретизация няма теоретични или организационни ограничения. Това обаче не може да се каже за високите скорости.

При разглеждания метод за организация на трансфера съществуват две възможности за постигане на пределно високи скорости на дискретизация и трансфер. Първата възможност се постига чрез повишаване честотата на стартиращия сигнал, което се изразява в намаляване на временния интервал **T3**. Теоретично този интервал може да се премахне, т.е. **T3=0**, но практически това е опасно да се прави. Опасността идва от факта, че края на временния интервал **T2** не винаги отстои на едно и също разстояние от началото на периода **T**. Това се дължи първо на асинхронността между тактовите импулси на сигнала **Start ADC** и тактовите импулси на самия преобразовател. Тази асинхронност въвежда променливото закъснение t_{ADC} (виж фиг.1). Второ, времето за което DMA-контролера заема системната шина t_{DMA} (виж фиг.1), е също променливо. В резултат на това съществува вероятност в някои от циклите на АЦ-контролера да се получи стартиране на преобразователя преди трансферът на текущия отчет да е завършил, т.е. $(T1+T2) > T$.

Втората възможност се състои в подходящия избор на аналого-цифров преобразовател. Колкото по-бързо той получава отчетите, толкова по-малък е временният интервал **T1**. Теоретично може да се приеме този интервал да клони към нула. Така излиза, че

скоростният предел за аналого-цифровото преобразуване се определя единствено от пропускателните възможности на системната шина. Като се имат предвид обаче особеностите на разглеждания режим за трансфер, при който в управлението на системната шина процесорът и DMA-контролерът се редуват, се стига до заключението, че минималната стойност към която може да се стреми периодът T на стартиращите импулси, е два машинни цикъла, което според стандарта ISA прави около $2\text{ }\mu\text{s}$. При използване на DMA-канал за 16-битов трансфер това означава натрупване на отчети със скорост близка до 10^6 [B/s] .

Ако аналого-цифровият преобразовател има изходен регистър за съхранение на данни, т.е. ако той може да съвместява във времето процедурите преобразуване и четене на данни, то тогава стартиращият преобразователя заден фронт на сигнала **Start ADC** може да се премества при определени условия още по-вляво от края на временния интервал T_2 , както е показано на фигура 2.



Фиг. 2

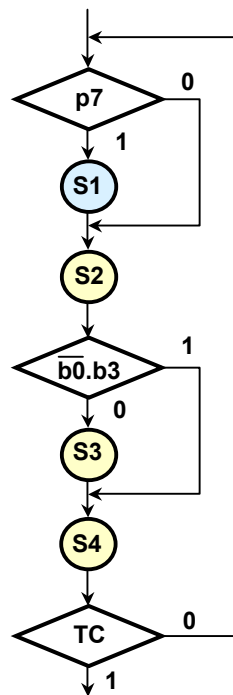
Условието, при което ще може да се работи по показания начин е: $T_1 > T_2$. Това означава, че описаните за този режим на работа методи за управление на АЦ-контролера, има смисъл да се прилагат само когато използваният преобразовател има време за преобразуване не по-малко от $1\text{ }\mu\text{s}$.

Б) Режим на блочен обмен

Когато DMA-каналът е програмиран за работа в този режим (битове b_7, b_6 в регистъра на режима имат стойностите $b_7=1, b_6=0$), трансферът на заявеното количество отчети се извършва чрез непрекъснато следващи цикли на обмен, инициирани от една единствена заявка. Веднъж заел системната шина DMA-контролерът не я освобождава докато не се формира флагът **TC** (*Terminal count*). При този начин на работа (непрекъснато следващи чисти цикли на обмен) DMA-контролерът разчита, че източникът на данни е винаги готов за четене. Тук това означава, че всеки отделен отчет АЦ-контролерът трябва да получава в рамките на един машинен цикъл. Скоростните параметри на преобразуването, изискванията към схемната част на контролера, както и алгоритмите за неговото управление най-точно се изясняват върху блок-схемата на микропрограмното управление на DMA-контролера [5], [6]. Тук на фигура 3 е представен обобщен граф на състоянията му при работа в блочен режим.

От фигурата се вижда, че ако не е изпълнено условието за край на цикъла за трансфер ($TC=0$), DMA-контролерът преминава последователно през четири състояния (S_1, S_2, S_3, S_4), при което издава необходимите управляващи сигнали към процесора, към паметта и към външното устройство. Но не във всеки цикъл тези състояния присъствуват едновременно. Обикновено преносът от осмия разряд на адреса е нула ($p_7=0$), което означава, че при нормална (некомпресирана) времедиаграма, се редуват състоянията (S_2, S_3, S_4), (S_2, S_3, S_4) Некомпресирана времедиаграма се изпълнява, когато битовете b_0 и b_3 в управляващия регистър на DMA-контролера образуват конюнкцията: $\overline{b_0} \cap b_3 = 0$. На всеки 256 такива цикъла (виж фиг. 3) възниква преносът p_7 ($p_7=1$) и тогава се извършва преход и през състояние S_1 - ... (S_2, S_3, S_4), (S_1, S_2, S_3, S_4), (S_2, S_3, S_4),... Тъй като интерес представлява пределната скорост за работа на АЦ-контролера, който следва да получава всеки отчет в рамките на един

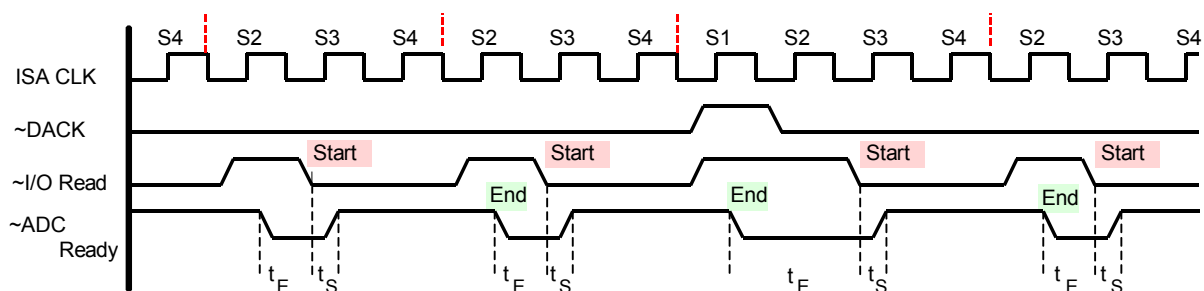
цикъл на DMA-контролера, от казаното следва, че типичното време за това е малко по-малко от три системни такта, т.е. $T_1 > 3 \cdot 210 = 630$ [ns].



Фиг. 3

За да се осигури този начин на работа всяко четене на данни от АЦ-контролера следва да осъществява старт на следващото преобразуване. За тази цел най-подходящ е сигналът **I/O Read**. На фигура 4 е представен процес на многократно четене на данни от АЦ-контролера, когато обслужващият го DMA-канал работи в режим на блочен обмен. Има се предвид също, че преобразователят съвместява във времето процесите на преобразуване и съхранение на предишния отчет.

Въпреки, че състоянията **Si** на DMA-контролера се сменят синхронно от кварцово стабилизирани тактови импулси, тази дискретизация ще определим като асинхронна. Потребителят трябва да знае, че временната стъпка между два съседни отчета, съхранявани в две съседни клетки на паметта, не винаги е една и съща - на всеки 256 стъпки от по 630 [ns] има една от 840 [ns]. Тя се отнася за отчети намиращи се в клетки, чиито адреси в младшия си байт преминават от стойност **xx...xxFF** в стойност **yy...yy00**. В случай, че за заснемане на реалния процес потребителят може да се задоволи с тези 256 отчета, то при подходящ начален адрес (**xx...xx00**), той ще ги получи при напълно синхронен процес. Ако се пренебрегне появата на състояние **S1** в машинния цикъл на всеки 256 такива от по три такта, може да се приеме, че при използване на 16-битов DMA-канал, работещ в режим на блочен обмен, е възможно натрупване на отчети със скорост близка до $3,17 \cdot 10^6$ [B/s].

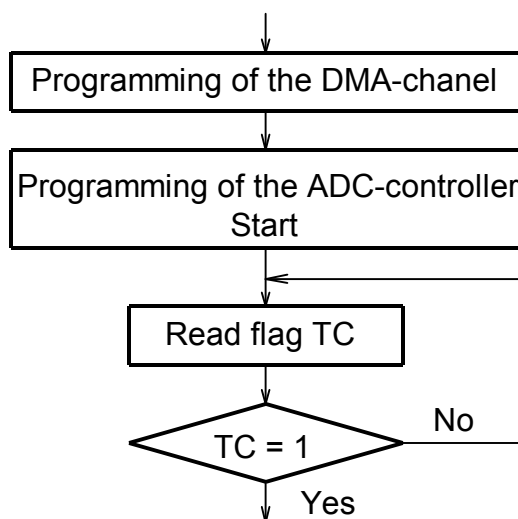


фиг. 4

Когато DMA-каналът е програмиран да реализира компресирана времедиаграма ($\overline{b0} \cap b3 = 1$), то от циклите на обмен се изключва състояние S3 (виж фигура 3). Тогава DMA-контролерът изпълнява своите цикли като преминава само през състояния S2 и S4, към които на всеки 256 повторения включва и състояние S1. Така типичното време на машинния цикъл се формира от два машинни такта, а скоростта за натрупване на отчети може да достигне стойности близки до $4,76 * 10^6$ [B/s]. Аналого-цифровият преобразовател, който може да осигури тази скорост, трябва да има време за преобразуване по-малко от 400 [ns].

2. Проблеми на синхронизацията

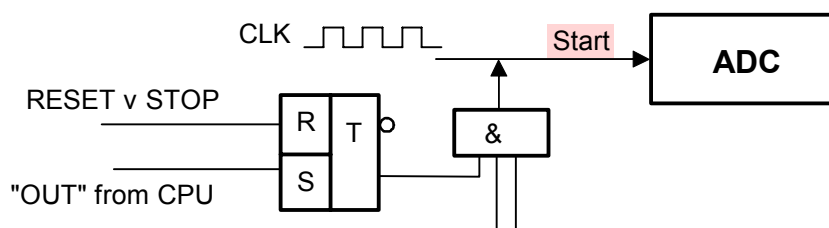
Управлението на един АЦ-контролер, базиран върху системната шина може да стане само с програмни средства. Ето защо проблемите на синхронизацията на програмата (т.е. процесора) с работата на контролера изискват пояснение. В изходно състояние АЦ-контролера е неактивен. Задействането му с цел получаване на данни от външен реален процес става чрез приложната програма в необходимия момент. Програмата следва да попадне в алгоритмичната структура, показана на фигура 5.



Фиг. 5

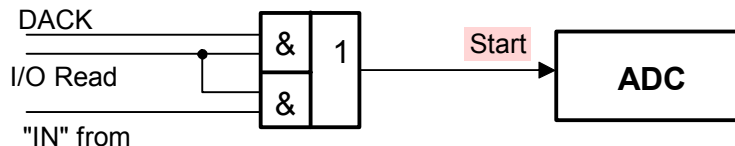
Условният блок за проверка на контролера реализира синхронизацията му с програмата. Независимо в кой режим DMA-каналът е извършил трансфера на данните, АЦ-контролерът е удобно да бъде изключен по чисто схемен път чрез сигнала **TC**, който DMA-контролерът разпространява по системната шина. Следователно синхронизацията може да се реализира чрез прочитане на съдържанието на регистъра на състоянието на DMA-контролера и анализ на стойността на флага **TC** за съответния DMA-канал.

Стартирането на аналого-цифровото преобразуване в АЦ-контролери, управлявани в режим на единичен обмен може да се постигне след разрешаване достъпа на тактовите импулси до преобразователя. За целта в програмата чрез команда **OUT** се записва инициализираща информация в съответния регистър на АЦ-контролера, на което съответствува примерната схема от фигура 6.



Фиг. 6

Стартирането на аналого-цифровото преобразуване в АЦ-контролери, управлявани в режим на блочен обмен не може да стане по описания статичен начин. Стартиращите импулси **I/O Read** в този режим (виж фиг. 4) се генерират в самия процес на обмен и ако той не бъде възбуден те няма откъде да се появят. Ето защо първият стартиращ импулс следва да се формира изкуствено от програмата. Това може да стане например чрез едно фиктивно четене на данни от регистър на АЦ-контролера. На казаното съответствува примерната схема от фигура 7.



Фиг. 7

Литература

- [1]. Койнаков Г.П., Кебеджиев А.Г., *Скоростна обработка на аналогови сигнали с компютър "Правец 16" по канала за пряк достъп до паметта*, "Автоматика и изчислителна техника" No 1, 1990.
- [2]. Тянев Д.С., *Двуканален АЦП-контролер с пряк достъп до паметта за компютри от тип IBM PC/XT*, НК "Дни на науката'92" Варна, X.1992.
- [3]. Tyanev D.S., Dobrev P.D., - *Two chanel 16-inputs A/D controller with DMA for IBM PC/AT*, 8-th International conference "Systems for automation of engineering and research", X.1994, Varna.
- [4]. Тянев Д.С., Атанасов А.Х., *Аналого-цифров контролер с програмируема честота на дискретизация с пряк достъп до паметта за компютри от тип IBM PC/AT*, Научни трудове на Русенски университет, Русе, X.1994.
- [5]. Yu-Cheng Liu, Glenn A. Gibson, *Microcomputer Systems: The 8086/8088 Family*, Prentice-Hall, 1984.
- [6]. Intel 80386 - hardware reference manual, Intel corp., Santa Clara, CA, 1986.