

# XFC – DISTRIBUTERADE KLOCKOR

XFC (eXtreme Fast Control) är Beckhoffs benämning på sin teknik för exakt tidssynkronisering i fältbussen EtherCAT, som hanterar tidsstämplar och distribuerade klockor.

Allmän information om XFC fungerar finns på hemsidan:

<http://www.beckhoff.se/english/highlights/xfc/>

Detaljerad information om funktion och handhavande finns i Beckhoff Information System:

[http://infosys.beckhoff.com/content/1033/ethercatsystem/html/bt\\_ethercat\\_dc\\_intro.htm?id=5132](http://infosys.beckhoff.com/content/1033/ethercatsystem/html/bt_ethercat_dc_intro.htm?id=5132)

## Teknisk bakgrund

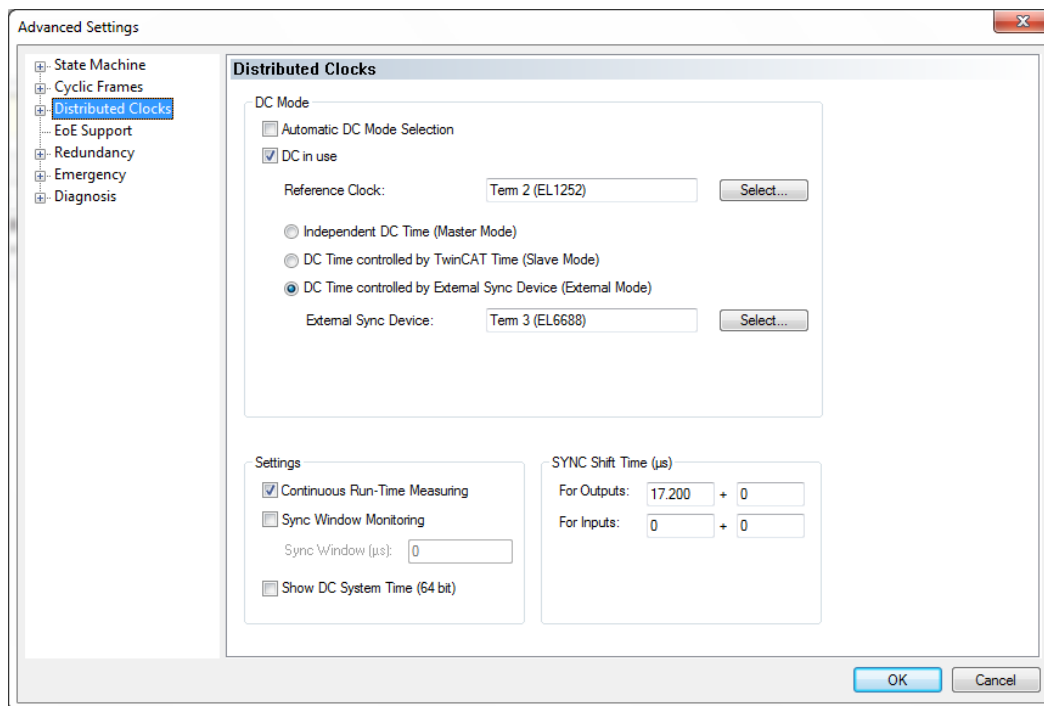
Detta dokument behandlar funktionsprincip och hantering av distribuerade klockor och tidsstämplar i TwinCAT med XFC-terminaler. Beskrivningar av handhavande är begränsade till enkla, digitala XFC-terminaler. Distribuerade klockor hänvisas ofta i texten som DC (Distributed Clocks).

# 1 Funktion

## 1.2 Allmänt

Samtliga XFC-terminaler har var sin intern klocka som fungerar individuellt. Dessa klockor synkroniseras mot en referensklocka för att kunna synkronisera I/O:n mycket exakt. Klockorna har en upplösning på en enda nanosekund. Precisionen i exaktheten som systemet kan hantera är på 100 nanosekunder. Denna exakta synkronisering kan exempelvis användas till synkronisering av utgångar eller tidsjämförelse av händelser från olika ingångar. Det är även användbart för att få en mer exakt positionskontroll, eliminera responstider eller kunna utföra en väldigt exakt tidsloggning. Dessutom kan en extern klocka användas som referensklocka för att kunna använda en verklig tid till synkroniseringen. Om man har behov att läsa in flera värden under en enda PLC-cykel så finns det XFC-terminaler med oversamlingsfunktion, de värden som terminalen får in mellan två olika busscykler lagras i ett internt register och skrivs till EtherCATens datagram när detta passerar terminalen. Har man behov av snabb fysisk respons i själva utgångarna så finns det XFC-terminaler med ”fast I/O”-funktion.

Standardinställningen i TwinCAT är att den första XFC-terminalen på EtherCAT-nätverket är referensklocka åt övriga DC-klockor i nätverket. Om det finns en terminal på nätverket som hanterar extern synkning så blir istället den referensklocka. Vid uppstarten av ett TwinCAT-baserat EtherCAT-system används TwinCAT-klockan som utgångspunkt för slaven med referensklocka om inget annat valts. Därefter fortsätter referensklockan att fungera individuellt om inte TwinCAT-tiden aktivt valts som referenstid.



Val av referensklocka i EtherCAT-masterns ”advanced settings”.

### 1.3 Synkronisering

Systemet måste ta hänsyn till den tidsmässiga fördröjningen för datautbyte mellan referensklocka och slavklocka varje slav får ta del av varje EtherCAT-frame. Detta sker genom att mastern under uppstartfasen skickar ut ett "Broadcast-Read-Datagram" till en speciell adress i varje ESC. Till denna skriver varje slav ner sin befintliga DC-tid vid dataöverföringen i respektive riktning. Mastern läser av tidsstämplarna och kan genom att beräkna skillnaderna göra en mycket exakt förutsägning av fördröjningar mellan slavarna.

Själva synkroniseringen av klockorna sker automatiskt och fortlöpande av mastern. Detta görs genom att ett särskilt datapaket, där slaven med referensklockan kan meddela sin aktuella tid, skickas över EtherCAT-nätverket och övriga slavklockor kan ta del av och synkronisera sig mot detta. Tack vare masterns sedan tidigare inlästa offset-värden, kan systemet göra en offset-kompensation innan värdet för en DC-tid presenteras.

### 1.4 DC:ns egenskaper i en EtherCAT Slave Controller

DC-klockan i varje XFC-slav har följande egenskaper:

- Enhet: 1 ns.
- Universal nollpunkt i datumformat: 2000-01-01 00:00
- 64 bitars räckvidd, vilket räcker till 584 år.

Varje slav synkroniseras med en noggrannhet på 100 ns, oavsett avstånd till master och övriga slavar. Den effektiva tidsenheten som varje klocka tickar i är 10 ns. "Entalet" i tidsangivelsen används i själva verket till styrning av slavklockorna.

Förutom tidsmässig synkronisering av in- och utgångar så kan DC-klockor hanteras med signaler utifrån:

**Compare – SYNC0/1:** En XFC-slav har oftast möjlighet till två olika typer av interrupt, SYNC0 och SYNC1. SYNC0 är en intern klockcykel i slaven som hanterar saker som sker mellan eller snabbare än själva fältbussens cykel. Om slavens DC-tid överensstämmer med en av användaren fördefinierad tid kan slaven verkställa ett avbrott och sammanhängande verksamhet bryts därmed. SYNC0-signalen hanterar även räknarfunktionen i räknarkort med DC-klockor.

En slav kan även läsa in nya standardvärden för att kunna genomföra en cyklisk sekvens av åtgärder (SYNC1). Exempelvis kan detta användas för att bestämma åtgärd när en SYNC0/1-signal påträffas. Detta kan vara aktuellt när oversamplingterminaler används.

**Capture – Latch 0/1:** En konfigurerad slav kan spara en aktuell DC-tid som en extern händelse inträffar utan dröjsmål i någon buffert. Exempel på externa händelser kan vara:

- Början på en EtherCAT-frame.
- Slutet på en EtherCAT-frame.
- En flank på en ingångssignal.
- Kommunikation mot en mikrokontroller.

## 2 Slavkonfiguration

### 2.1 DC-inställningar för en slav

Om en slav som har DC-klockor ingår i konfigurationen så har denna en flik benämnd ”DC” för inställningar av DC-klockor på slavnivå. Om en slav har flera olika driftlägen så kan dessa väljas här. De olika driftlägena avgör vad slavens interna arbetscykel ska synkroniseras efter. Denna arbetscykel avgör t.ex. när in- och utgångar sätts samt när minneskopieringar sker.

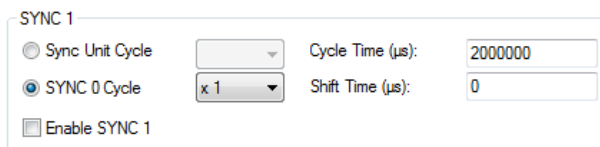
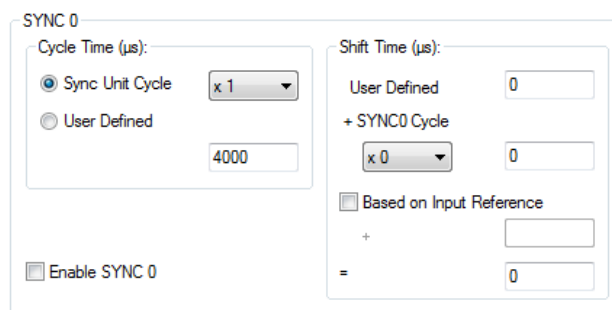
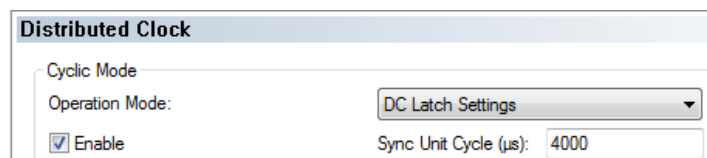
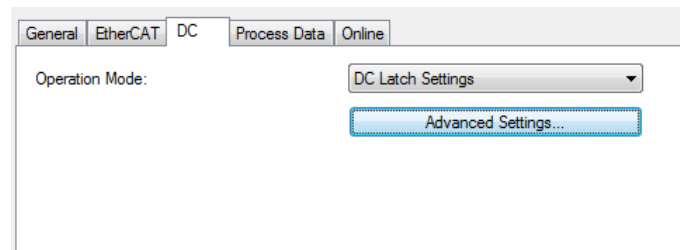
”Free run” innebär att slaven arbetar autonomt, utan synkronisering mot någon utomstående cykel. En del terminaler kan även ställas till SM-syncron och synkas då efter den ordinarie cykeltiden.

Under ”advanced settings” kan ytterligare val göras. Valet ”operation mode” är samma som i den överordnade fliken. ”Enable”-kryssrutan aktiverar och avaktiverar DC-klockor i slaven. ”Sync Unit Cycle” presenterar den grundcykeltid som slaven frågas av. Om slaven finns med i processdaten i fler än en task med olika cykeltider så är det den kortaste cykeltiden som anges här.

**Sync0-inställningar:** Här är ”Sync Unit Cycle” en multipel eller andel av den ovanstående grundcykeltiden. I denna intervall genereras SYNC0-signalen från aktuell slav.

”Shift Time” används för att förskjuta SYNC-pulsen en konstant tid framåt eller bakåt i relation till cykeltiden i mikrosekunder.

**Sync1-inställningar:** Sync1-cykeltiden kan vara en multipel eller bråkdel av antingen SYNC0-cykeltiden eller grundcykeln. Eventuell tidsförskjutning kan ställas på samma sätt här med ”Shift Time”.



## 3 Hantering i PLC-projekt

### 3.1 Datatyp

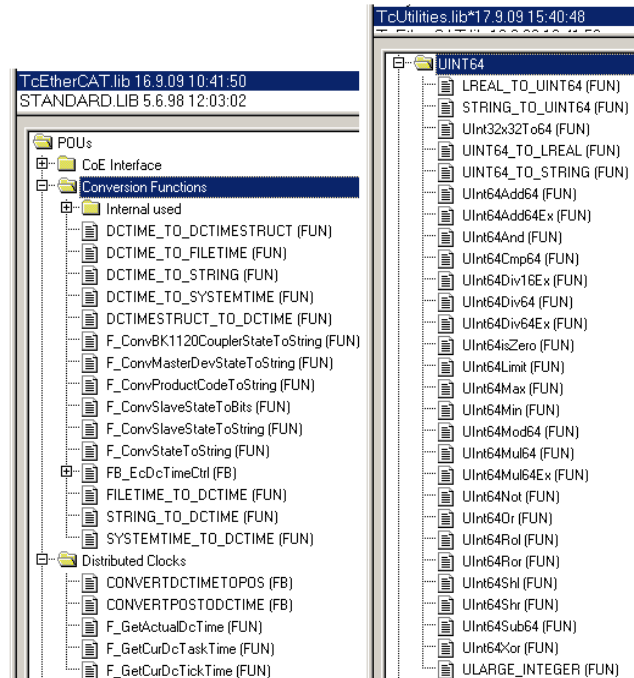
För att representera det stora tal av nanosekunder som DC-klockorna tickar, behöver en 64-bitsvariabel användas. Det finns tre olika datatyper som hanterar det:

**T DCTIME:** Baseras på

**T\_ULONG\_INTEGER** och är lämplig till att länkas till hårdvaruvariabler.

**T\_ULONG\_INTEGER:** Osignerad 64-bitars heltalstyp.

**T\_LONG\_INTEGER:** Signerad 64-bitars heltalstyp. Denna datatyp hanterar negativa värden, vilket kan vara nödvändigt vid mätning av tidsdifferanser, t.ex. vid extern tidssynkronisering (avhandlas ej i detta dokument).



*I TcEtherCAT-biblioteket finns det olika funktioner för att konvertera DC-tid till andra datatyper och i TcUtilities-biblioteket finns det diverse funktioner som hanterar 64-bitstal.*

### 3.2 Inläsning av DC-tid

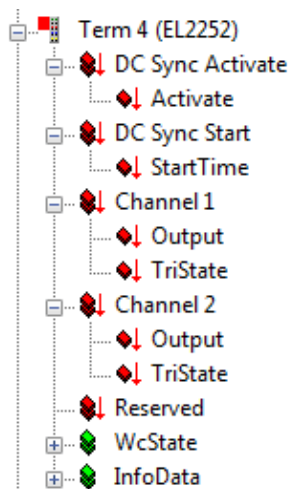
Det finns flera olika funktioner i TwinCAT som returnerar DC-klockornas tid, det vill säga ”nutid”:

**F\_GetActualDcTime:** Den faktiska befintliga DC-tiden. Kan anropas flera gånger i en task-cykel och ändå få olika värden.

**F\_GetCurDcTickTime:** Uppdateras när varje ”base time”-cykel påbörjas.

**F\_GetCurDcTaskTime:** Uppdateras varje task-cykel. Denna funktion rekommenderas att använda i första hand, då t.ex. slavarnas ”shift time” rättar sig efter denna.

### 3.3 Sätta en utgång med EL2252



- Activate: Byte som aktiverar terminalen att invänta starttiden "StartTime".
- StartTime: 64bit-värde som anger vid vilken DC-tid nästa utgångsswitchning ska ske.
- Output: Den fysiska utgången.

När heltalsvariabeln "activate" sätts från 0 till 3 börjar terminalen invänta att DC-klockorna uppnår den tid som "StartTime" satts till minst en taskcykel tidigare.

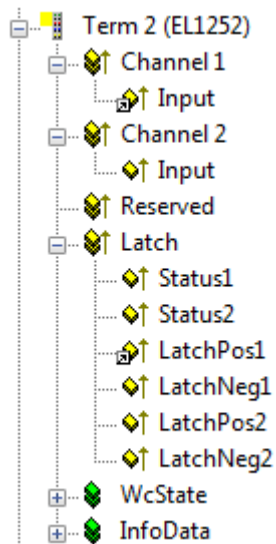
I nedanstående exempel för att sätta en utgång är de fetstilta variablerna länkade till motsvarande utgångar i terminalen. "startTime" är av datatypen T\_DcTime.

```

CASE iState OF
  0:
    currentTime := F_GetCurDcTaskTime();
    startTime := UInt64Add64(currentTime, ULARGE_INTEGER(0, 050_000_000));
    bOutput := TRUE;
    activate := 0;
    iState := 1;
  1:
    activate := 3;

```

### 3.4 Läs en ingång med EL1252



- Input: Status på den fysiska ingången.
- LatchPos: DC-tiden som ingången blir hög.
- LatchNeg: DC-tiden som ingången blir hög.
- Status: Bit 0 (pos) och 1 (neg) anger om en positiv eller negativ latch registrerats. Resettas genom att läsa av latchen.

Det räcker att länka en variabel av typen T\_DcTime till en latch för att läsa av tiden när en ingång blivit hög eller fallit. När latchen lästs av kan en ny tid registreras.

This publication contains statements about the suitability of our products for certain areas of application. These statements are based on typical features of our products. The examples shown in this publication are for demonstration purposes only. The information provided herein should not be regarded as specific operation characteristics. It is incumbent on the customer to check and decide whether a product is suit-able for use in a particular application. We do not give any warranty that the source code which is made available with this publication is complete or accurate. This publication may be changed at any time with-out prior notice. No liability is assumed for errors and/or omissions. Our products are described in detail in our data sheets and documentations. Product-specific warnings and cautions must be observed. For the latest version of our data sheets and documentations please visit our website ([www.beckhoff.com](http://www.beckhoff.com)).

© Beckhoff Automation GmbH, September 2009

The reproduction, distribution and utilization of this document as well as the communication of its contents to others without express authorization is prohibited. Offenders will be held liable for the payment of damages. All rights reserved in the event of the grant of a patent, utility model or design.