

Formalization of a Generalized Protocol for Clock Synchronization in Isabelle/HOL

Alwen Tiu
LORIA - <http://qsl.loria.fr>

December 12, 2009

Abstract

We formalize the generalized Byzantine fault-tolerant clock synchronization protocol of Schneider. This protocol abstracts from particular algorithms or implementations for clock synchronization. This abstraction includes several assumptions on the behaviors of physical clocks and on general properties of concrete algorithms/implementations. Based on these assumptions the correctness of the protocol is proved by Schneider. His proof was later verified by Shankar using the theorem prover EHDm (precursor to PVS). Our formalization in Isabelle/HOL is based on Shankar’s formalization.

Contents

1	Introduction	1
2	Isar proof scripts	2
2.1	Types and constants definitions	2
2.2	Clock conditions	3
2.2.1	Some derived properties of clocks	5
2.2.2	Bounded-drift for logical clocks (IC)	6
2.3	Agreement property	10

1 Introduction

In certain distributed systems, e.g., real-time process-control systems, the existence of a reliable global time source is critical in ensuring the correct functioning of the systems. This reliable global time source can be implemented using several physical clocks distributed on different nodes in the distributed system. Since physical clocks are by nature constantly drifting away from the “real time” and different clocks can have different drift rates, in such a scheme, it is important that these clocks are regularly adjusted so that they are closely synchronized within a certain application-specific safe bound. The design and verification of clock synchronization protocols are often complicated by the additional requirement that the protocols should work correctly under certain types of errors, e.g., failure of some clocks, error in communication network or corrupted messages, etc.

There has been a number of fault-tolerant clock synchronization algorithms studied in the literature, e.g., the *Interactive Convergence Algorithm* (ICA) by Lamport and Melliar-Smith [1], the Lundelius-Lynch algorithm [2], etc., each with its own degree of fault tolerance. One important property that

must be satisfied by a clock synchronization algorithm is the agreement property, i.e., at any time t , the difference of the clock readings of any two non-faulty processes must be bounded by a constant (which is fixed according to the domain of applications). At the core of these algorithms is the convergence function that calculates the adjustment to a clock of a process, based on the clock readings of all other processes. Schneider [3] gives an abstract characterization of a wide range of clock synchronization algorithms (based on the convergence functions used) and proves the agreement property in this abstract framework. Schneider’s proof was later verified by Shankar [4] in the theorem prover EHDM (precursor to PVS), where eleven axioms about clocks are explicitly stated.

We formalize Schneider’s proof in Isabelle/HOL, making use of Shankar’s formulation of the clock axioms. The particular formulation of axioms on clock conditions and the statements of the main theorems here are essentially those of Shankar’s [4], with some minor changes in syntax. For the full description of the protocol, the general structure of the proof and the meaning of the constants and function symbols used in this formalization, we refer readers to [4].

Acknowledgment I would like to thank Stephan Merz and Pascal Fontaine for useful tips on using Isabelle and particularly the Isar proof language.

2 Isar proof scripts

```
theory GenClock imports Complex-Main begin
```

2.1 Types and constants definitions

Process is represented by natural numbers. The type ‘event’ corresponds to synchronization rounds.

types

```
process = nat
event = nat
time = real
Clocktime = real
```

axiomatization

```
 $\delta$  :: real and
 $\mu$  :: real and
 $\rho$  :: real and
rmin :: real and
rmax :: real and
 $\beta$  :: real and
 $\Lambda$  :: real and
```

```
np :: process and
maxfaults :: process and
```

```
PC :: [process, time]  $\Rightarrow$  Clocktime and
```

```
VC :: [process, time]  $\Rightarrow$  Clocktime and
```

$te :: [process, event] \Rightarrow time$ **and**

$\vartheta :: [process, event] \Rightarrow (process \Rightarrow Clocktime)$ **and**

$IC :: [process, event, time] \Rightarrow Clocktime$ **and**

$correct :: [process, time] \Rightarrow bool$ **and**

$cf_n :: [process, (process \Rightarrow Clocktime)] \Rightarrow Clocktime$ **and**

$\pi :: [Clocktime, Clocktime] \Rightarrow Clocktime$ **and**

$\alpha :: Clocktime \Rightarrow Clocktime$

definition

$count :: [process \Rightarrow bool, process] \Rightarrow nat$ **where**
 $count\ f\ n = card\ \{p. p < n \wedge f\ p\}$

definition

$Adj :: [process, event] \Rightarrow Clocktime$ **where**
 $Adj = (\lambda\ p\ i. \text{if } 0 < i \text{ then } cf_n\ p\ (\vartheta\ p\ i) - PC\ p\ (te\ p\ i)$
 $\text{else } 0)$

definition

$okRead1 :: [process \Rightarrow Clocktime, Clocktime, process \Rightarrow bool] \Rightarrow bool$ **where**
 $okRead1\ f\ x\ ppred \iff (\forall\ l\ m. ppred\ l \wedge ppred\ m \longrightarrow |f\ l - f\ m| \leq x)$

definition

$okRead2 :: [process \Rightarrow Clocktime, process \Rightarrow Clocktime, Clocktime,$
 $process \Rightarrow bool] \Rightarrow bool$ **where**
 $okRead2\ f\ g\ x\ ppred \iff (\forall\ p. ppred\ p \longrightarrow |f\ p - g\ p| \leq x)$

definition

$\rho\text{-bound1} :: [[process, time] \Rightarrow Clocktime] \Rightarrow bool$ **where**
 $\rho\text{-bound1}\ C \iff (\forall\ p\ s\ t. correct\ p\ t \wedge s \leq t \longrightarrow C\ p\ t - C\ p\ s \leq (t - s) * (1 + \rho))$

definition

$\rho\text{-bound2} :: [[process, time] \Rightarrow Clocktime] \Rightarrow bool$ **where**
 $\rho\text{-bound2}\ C \iff (\forall\ p\ s\ t. correct\ p\ t \wedge s \leq t \longrightarrow (t - s) * (1 - \rho) \leq C\ p\ t - C\ p\ s)$

2.2 Clock conditions

Some general assumptions

axioms

$constants\text{-}ax: 0 < \beta \wedge 0 < \mu \wedge 0 < rmin$
 $\wedge rmin \leq rmax \wedge 0 < \rho \wedge 0 < np \wedge maxfaults \leq np$

$PC\text{-}monotone: \forall\ p\ s\ t. correct\ p\ t \wedge s \leq t \longrightarrow PC\ p\ s \leq PC\ p\ t$

$VClock: \forall\ p\ t\ i. correct\ p\ t \wedge te\ p\ i \leq t \wedge t < te\ p\ (i + 1) \longrightarrow VC\ p\ t = IC\ p\ i\ t$

$IClock: \forall\ p\ t\ i. correct\ p\ t \longrightarrow IC\ p\ i\ t = PC\ p\ t + Adj\ p\ i$

Condition 1: initial skew

axioms

$$\text{init}: \forall p. \text{correct } p \ 0 \longrightarrow 0 \leq PC \ p \ 0 \wedge PC \ p \ 0 \leq \mu$$

Condition 2: bounded drift

axioms

$$\text{rate-1}: \forall p \ s \ t. \text{correct } p \ t \wedge s \leq t \longrightarrow PC \ p \ t - PC \ p \ s \leq (t - s) * (1 + \varrho)$$

$$\text{rate-2}: \forall p \ s \ t. \text{correct } p \ t \wedge s \leq t \longrightarrow (t - s) * (1 - \varrho) \leq PC \ p \ t - PC \ p \ s$$

Condition 3: bounded interval

axioms

$$\text{rts0}: \forall p \ t \ i. \text{correct } p \ t \wedge t \leq te \ p \ (i+1) \longrightarrow t - te \ p \ i \leq rmax$$

$$\text{rts1}: \forall p \ t \ i. \text{correct } p \ t \wedge te \ p \ (i+1) \leq t \longrightarrow rmin \leq t - te \ p \ i$$

Condition 4 : bounded delay

axioms

$$\text{rts2a}: \forall p \ q \ t \ i. \text{correct } p \ t \wedge \text{correct } q \ t \wedge te \ q \ i + \beta \leq t \longrightarrow te \ p \ i \leq t$$

$$\text{rts2b}: \forall p \ q \ i. \text{correct } p \ (te \ p \ i) \wedge \text{correct } q \ (te \ q \ i) \longrightarrow \text{abs}(te \ p \ i - te \ q \ i) \leq \beta$$

Condition 5: initial synchronization

axioms

$$\text{synch0}: \forall p. te \ p \ 0 = 0$$

Condition 6: nonoverlap

axioms

$$\text{nonoverlap}: \beta \leq rmin$$

Condition 7: reading errors

axioms

$$\text{readerror}: \forall p \ q \ i. \text{correct } p \ (te \ p \ (i+1)) \wedge \text{correct } q \ (te \ p \ (i+1)) \longrightarrow \text{abs}(\vartheta \ p \ (i+1) \ q - IC \ q \ i \ (te \ p \ (i+1))) \leq \Lambda$$

Condition 8: bounded faults

axioms

$$\text{correct-closed}: \forall p \ s \ t. s \leq t \wedge \text{correct } p \ t \longrightarrow \text{correct } p \ s$$

$$\text{correct-count}: \forall t. np - maxfaults \leq \text{count} (\lambda p. \text{correct } p \ t) \ np$$

Condition 9: Translation invariance

axioms

$$\text{trans-inv}: \forall p \ f \ x. 0 \leq x \longrightarrow \text{cfn } p \ (\lambda y. f \ y + x) = \text{cfn } p \ f + x$$

Condition 10: precision enhancement

axioms

prec-enh:

$$\forall \text{ppred } p \ q \ f \ g \ x \ y.$$

$$np - maxfaults \leq \text{count } \text{ppred } np \wedge$$

$$\text{okRead1 } f \ y \ \text{ppred} \wedge \text{okRead1 } g \ y \ \text{ppred} \wedge$$

$$\text{okRead2 } f \ g \ x \ \text{ppred} \wedge \text{ppred } p \wedge \text{ppred } q$$

$$\longrightarrow \text{abs}(\text{cfn } p \ f - \text{cfn } q \ g) \leq \pi \ x \ y$$

Condition 11: accuracy preservation

axioms

acc-prsv:

\forall *ppred p q f x. okRead1 f x ppred* \wedge *np - maxfaults* \leq *count ppred np*
 \wedge *ppred p* \wedge *ppred q* \longrightarrow *abs(cfn p f - f q)* \leq α *x*

2.2.1 Some derived properties of clocks

lemma *rts0d:*

assumes *cp: correct p (te p (i+1))*
shows *te p (i+1) - te p i* \leq *rmax*
using *cp rts0* **by** *simp*

lemma *rts1d:*

assumes *cp: correct p (te p (i+1))*
shows *rmin* \leq *te p (i+1) - te p i*
using *cp rts1* **by** *simp*

lemma *rte:*

assumes *cp: correct p (te p (i+1))*
shows *te p i* \leq *te p (i+1)*

proof–

from *cp rts1d* **have** *rmin* \leq *te p (i+1) - te p i*
by *simp*

from *this constants-ax* **show** *?thesis* **by** *arith*

qed

lemma *beta-bound1:*

assumes *corr-p: correct p (te p (i+1))*
and *corr-q: correct q (te p (i+1))*
shows *0* \leq *te p (i+1) - te q i*

proof–

from *corr-p rte* **have** *te p i* \leq *te p (i+1)*
by *simp*

from *this corr-p correct-closed* **have** *corr-pi: correct p (te p i)*
by *blast*

from *corr-p rts1d nonoverlap* **have** *rmin* \leq *te p (i+1) - te p i*
by *simp*

from *this nonoverlap* **have** β \leq *te p (i+1) - te p i* **by** *simp*

hence *te p i + β* \leq *te p (i+1)* **by** *simp*

from *this corr-p corr-q rts2a*

have *te q i* \leq *te p (i+1)*

by *blast*

thus *?thesis* **by** *simp*

qed

lemma *beta-bound2:*

assumes *corr-p: correct p (te p (i+1))*
and *corr-q: correct q (te q i)*
shows *te p (i+1) - te q i* \leq *rmax + β*

proof–

from *corr-p rte* **have** *te p i* \leq *te p (i+1)*
by *simp*

```

from this corr-p correct-closed have corr-pi: correct p (te p i)
  by blast

have split: te p (i+1) - te q i =
  (te p (i+1) - te p i) + (te p i - te q i)
  by (simp)

from corr-q corr-pi rts2b have Eq1: abs(te p i - te q i) ≤ β
  by simp
have Eq2: te p i - te q i ≤ β
proof cases
  assume te q i ≤ te p i
  from this Eq1 show ?thesis
  by (simp add: real-abs-def)
next
  assume  $\neg$  (te q i ≤ te p i)
  from this Eq1 show ?thesis
  by (simp add: real-abs-def)
qed

from corr-p rts0d have te p (i+1) - te p i ≤ rmax
  by simp
from this split Eq2 show ?thesis by simp
qed

```

2.2.2 Bounded-drift for logical clocks (IC)

lemma *bd*:

```

assumes ie:  $s \leq t$ 
and rb1: rho-bound1 C
and rb2: rho-bound2 D
and PC-ie:  $D q t - D q s \leq C p t - C p s$ 
and corr-p: correct p t
and corr-q: correct q t
shows  $|C p t - D q t| \leq |C p s - D q s| + 2 * \rho * (t - s)$ 

```

proof–

```

let ?Dt =  $C p t - D q t$ 
let ?Ds =  $C p s - D q s$ 
let ?Bp =  $C p t - C p s$ 
let ?Bq =  $D q t - D q s$ 
let ?I =  $t - s$ 

```

have $|?Bp - ?Bq| \leq 2 * \rho * (t - s)$

proof–

```

from PC-ie have Eq1:  $|?Bp - ?Bq| = ?Bp - ?Bq$  by (simp add: real-abs-def)
from corr-p ie rb1 have Eq2:  $?Bp - ?Bq \leq ?I * (1 + \rho) - ?Bq$  (is  $?E1 \leq ?E2$ )
  by (simp add: diff-minus[symmetric] rho-bound1-def)
from corr-q ie rb2 have  $?I * (1 - \rho) \leq ?Bq$ 
  by (simp add: rho-bound2-def)
from this have Eq3:  $?E2 \leq ?I * (1 + \rho) - ?I * (1 - \rho)$ 
  by (simp)
have Eq4:  $?I * (1 + \rho) - ?I * (1 - \rho) = 2 * \rho * ?I$ 
  by (simp add: algebra-simps)

```

from $Eq1$ $Eq2$ $Eq3$ $Eq4$ **show** $?thesis$ **by** $simp$
qed
moreover
have $|?Dt| \leq |?Bp - ?Bq| + |?Ds|$
 by $(simp\ add:\ real-abs-def)$
ultimately show $?thesis$ **by** $simp$
qed

lemma *bounded-drift*:

assumes $ie: s \leq t$
and $rb1: rho-bound1\ C$
and $rb2: rho-bound2\ C$
and $rb3: rho-bound1\ D$
and $rb4: rho-bound2\ D$
and $corr-p: correct\ p\ t$
and $corr-q: correct\ q\ t$
shows $|C\ p\ t - D\ q\ t| \leq |C\ p\ s - D\ q\ s| + 2*\varrho*(t - s)$

proof–

let $?Bp = C\ p\ t - C\ p\ s$
let $?Bq = D\ q\ t - D\ q\ s$

show $?thesis$

proof *cases*

assume $?Bq \leq ?Bp$
from $this\ ie\ rb1\ rb4\ corr-p\ corr-q\ bd$ **show** $?thesis$ **by** $simp$
next
assume $\neg (?Bq \leq ?Bp)$
hence $?Bp \leq ?Bq$ **by** $simp$
from $this\ ie\ rb2\ rb3\ corr-p\ corr-q\ bd$
have $|D\ q\ t - C\ p\ t| \leq |D\ q\ s - C\ p\ s| + 2*\varrho*(t - s)$
 by $simp$
from $this$ **show** $?thesis$ **by** $(simp\ add:\ abs-minus-commute)$

qed

qed

Drift rate of logical clocks

lemma *IC-rate1*:

$rho-bound1\ (\lambda\ p\ t.\ IC\ p\ i\ t)$

proof–

{
fix $p::process$
fix $s::time$
fix $t::time$
assume $cp: correct\ p\ t$
assume $ie: s \leq t$
from $cp\ ie\ correct-closed$ **have** $cps: correct\ p\ s$
 by $blast$
have $IC\ p\ i\ t - IC\ p\ i\ s \leq (t - s)*(1 + \varrho)$
proof–
from $cp\ IClock$ **have** $IC\ p\ i\ t = PC\ p\ t + Adj\ p\ i$
 by $simp$
moreover
from $cps\ IClock$ **have** $IC\ p\ i\ s = PC\ p\ s + Adj\ p\ i$

```

    by simp
  moreover
  from cp ie rate-1 have PC p t - PC p s ≤ (t - s)*(1+ρ)
    by simp
  ultimately show ?thesis by simp
qed
}
thus ?thesis by (simp add: rho-bound1-def)
qed

```

lemma IC-rate2:

rho-bound2 ($\lambda p t. IC\ p\ i\ t$)

proof-

```

{
  fix p::process
  fix s::time
  fix t::time
  assume cp: correct p t
  assume ie: s ≤ t
  from cp ie correct-closed have cps: correct p s
    by blast
  have (t - s)*(1 - ρ) ≤ IC p i t - IC p i s
  proof-
    from cp IClock have IC p i t = PC p t + Adj p i
      by simp
    moreover
    from cps IClock have IC p i s = PC p s + Adj p i
      by simp
    moreover
    from cp ie rate-2 have (t - s)*(1-ρ) ≤ PC p t - PC p s
      by simp
    ultimately show ?thesis by simp
  qed
}
thus ?thesis by (simp add: rho-bound2-def)
qed

```

Auxiliary function *ICf*: we introduce this to avoid some unification problem in some tactic of isabelle.

definition

ICf :: nat ⇒ (process ⇒ time ⇒ Clocktime) **where**
ICf i = ($\lambda p t. IC\ p\ i\ t$)

lemma IC-bd:

assumes *ie*: $s \leq t$
and *corr-p*: correct p t
and *corr-q*: correct q t
shows $|IC\ p\ i\ t - IC\ q\ j\ t| \leq |IC\ p\ i\ s - IC\ q\ j\ s| + 2*\rho*(t - s)$

proof-

let ?C = *ICf* i
let ?D = *ICf* j
let ?G = $|?C\ p\ t - ?D\ q\ t| \leq |?C\ p\ s - ?D\ q\ s| + 2*\rho*(t - s)$

from *IC-rate1* **have** *rb1*: *rho-bound1* (*ICf i*) \wedge *rho-bound1* (*ICf j*)
by (*simp add: ICf-def*)

from *IC-rate2* **have** *rb2*: *rho-bound2* (*ICf i*) \wedge *rho-bound2* (*ICf j*)
by (*simp add: ICf-def*)

from *ie rb1 rb2 corr-p corr-q bounded-drift*
have *?G* **by** *simp*

from *this* **show** *?thesis* **by** (*simp add: ICf-def*)
qed

lemma *event-bound*:

assumes *ie1*: $0 \leq (t::real)$

and *corr-p*: *correct p t*

and *corr-q*: *correct q t*

shows $\exists i. t < \max (te\ p\ i) (te\ q\ i)$

proof (*rule ccontr*)

assume *A*: $\neg (\exists i. t < \max (te\ p\ i) (te\ q\ i))$

show *False*

proof–

have *F1*: $\forall i. te\ p\ i \leq t$

proof

fix *i* :: *nat*

from *A* **have** $\neg (t < \max (te\ p\ i) (te\ q\ i))$

by *simp*

hence *Eq1*: $\max (te\ p\ i) (te\ q\ i) \leq t$ **by** *arith*

have *Eq2*: $te\ p\ i \leq \max (te\ p\ i) (te\ q\ i)$

by (*simp add: max-def*)

from *Eq1 Eq2* **show** $te\ p\ i \leq t$ **by** *simp*

qed

have *F2*: $\forall (i :: nat). \text{correct } p (te\ p\ i)$

proof

fix *i* :: *nat*

from *F1* **have** $te\ p\ i \leq t$ **by** *simp*

from *this* *corr-p correct-closed*

show *correct p (te p i)* **by** *blast*

qed

have *F3*: $\forall (i :: nat). \text{real } i * rmin \leq te\ p\ i$

proof

fix *i* :: *nat*

show $\text{real } i * rmin \leq te\ p\ i$

proof (*induct i*)

from *synch0* **show** $\text{real } (0::nat) * rmin \leq te\ p\ 0$ **by** *simp*

next

fix *i* :: *nat* **assume** *ind-hyp*: $\text{real } i * rmin \leq te\ p\ i$

show $\text{real } (Suc\ i) * rmin \leq te\ p\ (Suc\ i)$

proof–

have *Eq1*: $\text{real } i * rmin + rmin = (\text{real } i + 1) * rmin$

by (simp add: real-add-mult-distrib)
 have Eq2: $\text{real } i + 1 = \text{real } (i+1)$ by simp
 from Eq1 Eq2
 have Eq3: $\text{real } i * rmin + rmin = \text{real } (i+1) * rmin$
 by (simp)

from F2 have cp1: correct p (te p (i+1))
 by simp
 from F2 have cp2: correct p (te p i)
 by simp
 from cp1 rts1d have $rmin \leq \text{te } p (i+1) - \text{te } p i$
 by simp
 hence Eq4: $\text{te } p i + rmin \leq \text{te } p (i+1)$ by simp
 from ind-hyp have $\text{real } i * rmin + rmin \leq \text{te } p i + rmin$
 by (simp)
 from this Eq4 have $\text{real } i * rmin + rmin \leq \text{te } p (i+1)$
 by simp
 from this Eq3 show ?thesis by simp
 qed
 qed
 qed

have F4: $\forall (i::\text{nat}). \text{real } i * rmin \leq t$
 proof
 fix i::nat
 from F1 have $\text{te } p i \leq t$ by simp
 moreover
 from F3 have $\text{real } i * rmin \leq \text{te } p i$ by simp
 ultimately show $\text{real } i * rmin \leq t$ by simp
 qed

from constants-ax have $0 < rmin$ by simp

from this reals-Archimedean3
 have Archi: $\exists (k::\text{nat}). t < \text{real } k * rmin$
 by blast

from Archi obtain $k::\text{nat}$ where $C: t < \text{real } k * rmin$..

from F4 have $\text{real } k * rmin \leq t$ by simp
 hence notC: $\neg (t < \text{real } k * rmin)$ by simp

from C notC show False by simp
 qed
 qed

2.3 Agreement property

definition $\gamma1 x = \pi (2 * \rho * \beta + 2 * \Lambda) (2 * \Lambda + x + 2 * \rho * (rmax + \beta))$

definition $\gamma2 x = x + 2 * \rho * rmax$

definition $\gamma3 x = \alpha (2 * \Lambda + x + 2 * \rho * (rmax + \beta)) + \Lambda + 2 * \rho * \beta$

definition

$okmaxsync :: [nat, Clocktime] \Rightarrow bool$ **where**
 $okmaxsync\ i\ x \longleftrightarrow (\forall\ p\ q. correct\ p\ (max\ (te\ p\ i)\ (te\ q\ i))$
 $\wedge\ correct\ q\ (max\ (te\ p\ i)\ (te\ q\ i)) \longrightarrow$
 $|IC\ p\ i\ (max\ (te\ p\ i)\ (te\ q\ i)) - IC\ q\ i\ (max\ (te\ p\ i)\ (te\ q\ i))| \leq x)$

definition

$okClocks :: [process, process, nat] \Rightarrow bool$ **where**
 $okClocks\ p\ q\ i \longleftrightarrow (\forall\ t. 0 \leq t \wedge t < max\ (te\ p\ i)\ (te\ q\ i)$
 $\wedge\ correct\ p\ t \wedge correct\ q\ t$
 $\longrightarrow |VC\ p\ t - VC\ q\ t| \leq \delta)$

lemma *okClocks-sym:*

assumes *ok-pq:* $okClocks\ p\ q\ i$

shows $okClocks\ q\ p\ i$

proof-

$\{$
 $\text{fix } t :: time$
 $\text{assume } ie1: 0 \leq t$
 $\text{assume } ie2: t < max\ (te\ q\ i)\ (te\ p\ i)$
 $\text{assume } corr-q: correct\ q\ t$
 $\text{assume } corr-p: correct\ p\ t$

 $\text{have } max\ (te\ q\ i)\ (te\ p\ i) = max\ (te\ p\ i)\ (te\ q\ i)$
 $\text{by } (simp\ add: max-def)$
 $\text{from } this\ ok-pq\ ie1\ ie2\ corr-p\ corr-q$
 $\text{have } |VC\ q\ t - VC\ p\ t| \leq \delta$
 $\text{by } (simp\ add: abs-minus-commute\ okClocks-def)$
 $\}$
 $\text{thus } ?thesis\ \text{by } (simp\ add: okClocks-def)$

qed

lemma *ICp-Suc:*

assumes *corr-p:* $correct\ p\ (te\ p\ (i+1))$

shows $IC\ p\ (i+1)\ (te\ p\ (i+1)) = cfn\ p\ (\vartheta\ p\ (i+1))$

using *corr-p* *IClock* **by**(*simp add: Adj-def*)

lemma *IC-trans-inv:*

assumes *ie1:* $te\ q\ (i+1) \leq te\ p\ (i+1)$

and *corr-p:* $correct\ p\ (te\ p\ (i+1))$

and *corr-q:* $correct\ q\ (te\ p\ (i+1))$

shows

$IC\ q\ (i+1)\ (te\ p\ (i+1)) =$
 $cfn\ q\ (\lambda\ n. \vartheta\ q\ (i+1)\ n + (PC\ q\ (te\ p\ (i+1)) - PC\ q\ (te\ q\ (i+1))))$
(is ?T1 = ?T2)

proof-

let $?X = PC\ q\ (te\ p\ (i+1)) - PC\ q\ (te\ q\ (i+1))$

from *corr-q* *ie1* *PC-monotone* **have** $posX: 0 \leq ?X$

by (*simp add: le-diff-eq*)

from *IClock* *corr-q* **have** $?T1 = cfn\ q\ (\vartheta\ q\ (i+1)) + ?X$

by(*simp add: Adj-def*)

from *this posX trans-inv* show *?thesis* by *simp*
qed

lemma *beta-rho*:

assumes *ie*: $te\ q\ (i+1) \leq te\ p\ (i+1)$

and *corr-p*: *correct p (te p (i+1))*

and *corr-q*: *correct q (te p (i+1))*

and *corr-l*: *correct l (te p (i+1))*

shows $|(PC\ l\ (te\ p\ (i+1)) - PC\ l\ (te\ q\ (i+1))) - (te\ p\ (i+1) - te\ q\ (i+1))| \leq \beta * \varrho$

proof-

let $?X = (PC\ l\ (te\ p\ (i+1)) - PC\ l\ (te\ q\ (i+1)))$

let $?D = te\ p\ (i+1) - te\ q\ (i+1)$

from *ie* have *posD*: $0 \leq ?D$ by *simp*

from *ie PC-monotone corr-l* have *posX*: $0 \leq ?X$

by (*simp add: le-diff-eq*)

from *ie corr-l rate-1* have *bound1*: $?X \leq ?D * (1 + \varrho)$ by *simp*

from *ie corr-l correct-closed* have *corr-l-tq*: *correct l (te q (i+1))*

by (*blast*)

from *ie corr-q correct-closed* have *corr-q-tq*: *correct q (te q (i+1))*

by *blast*

from *corr-q-tq corr-p rts2b* have $|?D| \leq \beta$

by (*simp*)

from *this constants-ax posD* have *D-beta*: $?D * \varrho \leq \beta * \varrho$

by (*simp add: real-abs-def*)

show *?thesis*

proof *cases*

assume *A*: $?D \leq ?X$

from *posX posD A* have *absEq*: $|?X - ?D| = ?X - ?D$

by (*simp add: real-abs-def*)

from *bound1* have *bound2*: $?X - ?D \leq ?D * \varrho$

by (*simp add: real-mult-commute real-add-mult-distrib*)

from *D-beta absEq bound2* show *?thesis* by *simp*

next

assume *notA*: $\neg (?D \leq ?X)$

from *this* have *absEq2*: $|?X - ?D| = ?D - ?X$

by (*simp add: real-abs-def*)

from *ie corr-l rate-2* have *bound3*: $?D * (1 - \varrho) \leq ?X$ by *simp*

from *this* have $?D - ?X \leq ?D * \varrho$ by (*simp add: algebra-simps*)

from *this absEq2 D-beta* show *?thesis* by *simp*

qed

qed

This lemma (and the next one *pe-cond2*) proves an assumption used in the precision enhancement.

lemma *pe-cond1*:

assumes *ie*: $te\ q\ (i+1) \leq te\ p\ (i+1)$

and *corr-p*: *correct p (te p (i+1))*

and *corr-q*: *correct q (te p (i+1))*

and *corr-l*: *correct l (te p (i+1))*

shows $|\vartheta\ q\ (i+1)\ l + (PC\ q\ (te\ p\ (i+1)) - PC\ q\ (te\ q\ (i+1))) -$

$\vartheta\ p\ (i+1)\ l| \leq 2 * \varrho * \beta + 2 * \Lambda$

(is $?M \leq ?N$)

proof-

let $?Xl = (PC\ l\ (te\ p\ (i+1)) - PC\ l\ (te\ q\ (i+1)))$
let $?Xq = (PC\ q\ (te\ p\ (i+1)) - PC\ q\ (te\ q\ (i+1)))$
let $?D = te\ p\ (i+1) - te\ q\ (i+1)$
let $?T = \vartheta\ p\ (i+1)\ l - \vartheta\ q\ (i+1)\ l$
let $?RE1 = \vartheta\ p\ (i+1)\ l - IC\ l\ i\ (te\ p\ (i+1))$
let $?RE2 = \vartheta\ q\ (i+1)\ l - IC\ l\ i\ (te\ q\ (i+1))$
let $?ICT = IC\ l\ i\ (te\ p\ (i+1)) - IC\ l\ i\ (te\ q\ (i+1))$

have $?M = |(?Xq - ?D) - (?T - ?D)|$
by(*simp add: real-abs-def*)

hence *Split*: $?M \leq |?Xq - ?D| + |?T - ?D|$
by(*simp add: real-abs-def*)

from *ie corr-q correct-closed* have *corr-q-tq: correct q (te q (i+1))*
by(*blast*)
from *ie corr-l correct-closed* have *corr-l-tq: correct l (te q (i+1))*
by *blast*

from *corr-p corr-q corr-l ie beta-rho*
have *XlD*: $|?Xl - ?D| \leq \beta * \varrho$
by *simp*

from *corr-p corr-q ie beta-rho*
have *XqD*: $|?Xq - ?D| \leq \beta * \varrho$ by *simp*

have *TD*: $|?T - ?D| \leq 2 * \Lambda + \beta * \varrho$

proof-

have *Eq1*: $|?T - ?D| = |(?T - ?ICT) + (?ICT - ?D)|$ (is $?E1 = ?E2$)
by (*simp add: real-abs-def*)

have *Eq2*: $?E2 \leq |?T - ?ICT| + |?ICT - ?D|$
by(*simp add: real-abs-def*)

have *Eq3*: $|?T - ?ICT| \leq |?RE1| + |?RE2|$
by(*simp add: real-abs-def*)

from *readerror corr-p corr-l*
have *Eq4*: $|?RE1| \leq \Lambda$ by *simp*

from *corr-l-tq corr-q-tq this readerror*
have *Eq5*: $|?RE2| \leq \Lambda$ by *simp*

from *Eq3 Eq4 Eq5* have *Eq6*: $|?T - ?ICT| \leq 2 * \Lambda$
by *simp*

have *Eq7*: $?ICT - ?D = ?Xl - ?D$

proof-

from *corr-p rte* have $te\ p\ i \leq te\ p\ (i+1)$
by(*simp*)

from *this corr-l correct-closed* **have** *corr-l-tpi: correct l (te p i)*
by *blast*
from *corr-q-tq rte* **have** *te q i ≤ te q (i+1)*
by *simp*
from *this corr-l-tq correct-closed* **have** *corr-l-tqi: correct l (te q i)*
by *blast*

from *IClock corr-l*
have *F1: IC l i (te p (i+1)) = PC l (te p (i+1)) + Adj l i*
by(*simp*)
from *IClock corr-l-tq*
have *F2: IC l i (te q (i+1)) = PC l (te q (i+1)) + Adj l i*
by *simp*
from *F1 F2* **show** *?thesis* **by**(*simp*)

qed

from *this XlD* **have** *Eq8: |?ICT - ?D| ≤ β*ρ*
by *arith*
from *Eq1 Eq2 Eq6 Eq8* **show** *?thesis*
by(*simp*)

qed

from *Split XqD TD* **have** *F1: ?M ≤ 2* β * ρ + 2*Λ*
by(*simp*)
have *F2: 2 * ρ * β + 2*Λ = 2* β * ρ + 2*Λ*
by *simp*
from *F1* **show** *?thesis* **by** (*simp only: F2*)

qed

lemma *pe-cond2:*

assumes *ie: te m i ≤ te l i*
and *corr-k: correct k (te k (i+1))*
and *corr-l-tk: correct l (te k (i+1))*
and *corr-m-tk: correct m (te k (i+1))*
and *ind-hyp: |IC l i (te l i) - IC m i (te l i)| ≤ δS*
shows $|\vartheta k (i+1) l - \vartheta k (i+1) m| \leq 2*\Lambda + \delta S + 2*\rho*(rmax + \beta)$

proof–

let *?X = ϑ k (i+1) l - ϑ k (i+1) m*
let *?N = 2*Λ + δS + 2*ρ*(rmax + β)*
let *?D1 = ϑ k (i+1) l - IC l i (te k (i+1))*
let *?D2 = ϑ k (i+1) m - IC m i (te k (i+1))*
let *?ICS = IC l i (te k (i+1)) - IC m i (te k (i+1))*
let *?tlm = te l i*
let *?IC = IC l i ?tlm - IC m i ?tlm*

have *Eq1: |?X| = |(?D1 - ?D2) + ?ICS|* (**is** *?E1 = ?E2*)
by (*simp add: real-abs-def*)

have *Eq2: ?E2 ≤ |?D1 - ?D2| + |?ICS|* **by** (*simp add: real-abs-def*)

from *corr-l-tk corr-k beta-bound1* **have** *ie-lk: te l i ≤ te k (i+1)*
by (*simp add: le-diff-eq*)

from *this corr-l-tk correct-closed* **have** *corr-l: correct l (te l i)*
by *blast*

from *ie-lk corr-l-tk corr-m-tk IC-bd*
have *Eq3: $|?ICS| \leq |?IC| + 2*\rho*(te\ k\ (i+1) - ?tIm)$*
by *simp*

from *this ind-hyp* **have** *Eq4: $|?ICS| \leq \delta S + 2*\rho*(te\ k\ (i+1) - ?tIm)$*
by *simp*

from *corr-l corr-k beta-bound2* **have** *$te\ k\ (i+1) - ?tIm \leq rmax + \beta$*
by *simp*

from *this constants-ax* **have** *$2*\rho*(te\ k\ (i+1) - ?tIm) \leq 2*\rho*(rmax + \beta)$*
by *(simp add: real-mult-le-cancel-iff2)*

from *this Eq4* **have** *Eq4a: $|?ICS| \leq \delta S + 2*\rho*(rmax + \beta)$*
by *(simp)*

from *corr-k corr-l-tk readerror*
have *Eq5: $|?D1| \leq \Lambda$* **by** *simp*

from *corr-k corr-m-tk readerror*
have *Eq6: $|?D2| \leq \Lambda$* **by** *simp*

have *$|?D1 - ?D2| \leq |?D1| + |?D2|$* **by** *(simp add: real-abs-def)*

from *this Eq5 Eq6* **have** *Eq7: $|?D1 - ?D2| \leq 2*\Lambda$*
by *(simp)*

from *Eq1 Eq2 Eq4a Eq7 split* **show** *?thesis* **by** *(simp)*

qed

lemma *theta-bound:*

assumes *corr-l: correct l (te p (i+1))*
and *corr-m: correct m (te p (i+1))*
and *corr-p: correct p (te p (i+1))*
and *IC-bound:*

$|IC\ l\ i\ (max\ (te\ l\ i)\ (te\ m\ i)) - IC\ m\ i\ (max\ (te\ l\ i)\ (te\ m\ i))|$
 $\leq \delta S$

shows $|\vartheta\ p\ (i+1)\ l - \vartheta\ p\ (i+1)\ m|$
 $\leq 2*\Lambda + \delta S + 2*\rho*(rmax + \beta)$

proof-

from *corr-p corr-l beta-bound1* **have** *tli-le-tp: $te\ l\ i \leq te\ p\ (i+1)$*
by *(simp add: le-diff-eq)*

from *corr-p corr-m beta-bound1* **have** *tmi-le-tp: $te\ m\ i \leq te\ p\ (i+1)$*
by *(simp add: le-diff-eq)*

let *?tml = $max\ (te\ l\ i)\ (te\ m\ i)$*

from *tli-le-tp tmi-le-tp* **have** *tml-le-tp: $?tml \leq te\ p\ (i+1)$*
by *simp*

from *tml-le-tp corr-l correct-closed* **have** *corr-l-tml: correct l ?tml*
by *blast*

from *tml-le-tp corr-m correct-closed* **have** *corr-m-tml: correct m ?tml*
by *blast*

let *?Y = $2*\Lambda + \delta S + 2*\rho*(rmax + \beta)$*

show $|\vartheta p (i+1) l - \vartheta p (i+1) m| \leq ?Y$
proof *cases*
 assume $A: te m i < te l i$

 from *this IC-bound*
 have $|IC l i (te l i) - IC m i (te l i)| \leq \delta S$
 by (*simp add: max-def*)
 from *this A corr-p corr-l corr-m pe-cond2*
 show *?thesis* **by** (*simp*)
next
 assume $\neg (te m i < te l i)$
 hence *Eq1: te l i ≤ te m i* **by** *simp*
 from *this IC-bound*
 have *Eq2: |IC l i (te m i) - IC m i (te m i)| ≤ δS*
 by (*simp add: max-def*)

 hence $|IC m i (te m i) - IC l i (te m i)| \leq \delta S$
 by (*simp add: abs-minus-commute*)
 from *this Eq1 corr-p corr-l corr-m pe-cond2*
 have $|\vartheta p (i+1) m - \vartheta p (i+1) l| \leq ?Y$
 by (*simp*)
 thus *?thesis* **by** (*simp add: abs-minus-commute*)
qed
qed

lemma *four-one-ind-half*:
 assumes $ie1: \beta \leq rmin$
 and $ie2: \mu \leq \delta S$
 and $ie3: \gamma1 \delta S \leq \delta S$
 and *ind-hyp: okmaxsync i δS*
 and $ie4: te q (i+1) \leq te p (i+1)$
 and *corr-p: correct p (te p (i+1))*
 and *corr-q: correct q (te p (i+1))*
shows $|IC p (i+1) (te p (i+1)) - IC q (i+1) (te p (i+1))| \leq \delta S$
proof–
 let $?tpq = te p (i+1)$

 let $?f = \lambda n. \vartheta q (i+1) n + (PC q (te p (i+1)) - PC q (te q (i+1)))$
 let $?g = \vartheta p (i+1)$

 from *ie4 corr-q correct-closed* **have** *corr-q-tq: correct q (te q (i+1))*
 by *blast*

 have *Eq-IC-cfn: |IC p (i+1) ?tpq - IC q (i+1) ?tpq| =*
 $|cfn q ?f - cfn p ?g|$
proof–
 from *corr-p ICp-Suc* **have** *Eq1: IC p (i+1) ?tpq = cfn p ?g* **by** *simp*

 from *ie4 corr-p corr-q IC-trans-inv*
 have *Eq2: IC q (i+1) ?tpq = cfn q ?f* **by** *simp*

 from *Eq1 Eq2* **show** *?thesis* **by** (*simp add: real-abs-def*)
qed

```

let ?ppred = λ l. correct l (te p (i+1))

let ?X = 2*ρ*β + 2*Λ
have ∀ l. ?ppred l → |?f l - ?g l| ≤ ?X
proof -
  {
    fix l
    assume ?ppred l
    from ie4 corr-p corr-q this pe-cond1
    have |?f l - ?g l| ≤ (2*ρ*β + 2*Λ)
      by(auto)
  }
  thus ?thesis by blast
qed
hence cond1: okRead2 ?f ?g ?X ?ppred
  by(simp add: okRead2-def)

let ?Y = 2*Λ + δS + 2*ρ*(rmax + β)

have ∀ l m. ?ppred l ∧ ?ppred m → |?f l - ?f m| ≤ ?Y
proof-
  {
    fix l m
    assume corr-l: ?ppred l
    assume corr-m: ?ppred m

    from corr-p corr-l beta-bound1 have tli-le-tp: te l i ≤ te p (i+1)
      by (simp add: le-diff-eq)
    from corr-p corr-m beta-bound1 have tmi-le-tp: te m i ≤ te p (i+1)
      by (simp add: le-diff-eq)

    let ?tlm = max (te l i) (te m i)

    from tli-le-tp tmi-le-tp have tlm-le-tp: ?tlm ≤ te p (i+1)
      by simp

    from ie4 corr-l correct-closed have corr-l-tq: correct l (te q (i+1))
      by blast
    from ie4 corr-m correct-closed have corr-m-tq: correct m (te q (i+1))
      by blast
    from tlm-le-tp corr-l correct-closed have corr-l-tlm: correct l ?tlm
      by blast
    from tlm-le-tp corr-m correct-closed have corr-m-tlm: correct m ?tlm
      by blast

    from ind-hyp corr-l-tlm corr-m-tlm
    have EqAbs1: |IC l i ?tlm - IC m i ?tlm| ≤ δS
      by(auto simp add: okmaxsync-def)

    have EqAbs3: |?f l - ?f m| = |ϑ q (i+1) l - ϑ q (i+1) m|
      by (simp add: real-abs-def)
  }

```

from *EqAbs1 corr-q-tq corr-l-tq corr-m-tq theta-bound*
have $|\vartheta q (i+1) l - \vartheta q (i+1) m| \leq ?Y$
by *simp*
from *this EqAbs3* **have** $|\?f l - \?f m| \leq ?Y$
by *simp*
}
thus *?thesis* **by** *simp*
qed
hence *cond2a: okRead1 ?f ?Y ?ppred* **by** (*simp add: okRead1-def*)

have $\forall l m. \?ppred l \wedge \?ppred m \longrightarrow |\?g l - \?g m| \leq ?Y$
proof-
{
fix *l m*
assume *corr-l: ?ppred l*
assume *corr-m: ?ppred m*

from *corr-p corr-l beta-bound1* **have** *tli-le-tp: te l i ≤ te p (i+1)*
by (*simp add: le-diff-eq*)
from *corr-p corr-m beta-bound1* **have** *tmi-le-tp: te m i ≤ te p (i+1)*
by (*simp add: le-diff-eq*)

let *?tlm = max (te l i) (te m i)*
from *tli-le-tp tmi-le-tp* **have** *tlm-le-tp: ?tlm ≤ te p (i+1)*
by *simp*

from *tlm-le-tp corr-l correct-closed* **have** *corr-l-tlm: correct l ?tlm*
by *blast*
from *tlm-le-tp corr-m correct-closed* **have** *corr-m-tlm: correct m ?tlm*
by *blast*

from *ind-hyp corr-l-tlm corr-m-tlm*
have *EqAbs1: |IC l i ?tlm - IC m i ?tlm| ≤ δS*
by (*auto simp add: okmaxsync-def*)

from *EqAbs1 corr-p corr-l corr-m theta-bound*
have $|\?g l - \?g m| \leq ?Y$ **by** *simp*
}
thus *?thesis* **by** *simp*
qed
hence *cond2b: okRead1 ?g ?Y ?ppred* **by** (*simp add: okRead1-def*)

from *correct-count* **have** $np - \text{maxfaults} \leq \text{count } \?ppred \ np$
by *simp*
from *this corr-p corr-q cond1 cond2a cond2b prec-enh*
have $|\text{cfn } q \?f - \text{cfn } p \?g| \leq \pi \?X \?Y$
by *blast*

from *ie3 this* **have** $|\text{cfn } q \?f - \text{cfn } p \?g| \leq \delta S$
by (*simp add: γ1-def*)

from *this Eq-IC-cfn* **show** *?thesis* **by** (*simp*)
qed

Theorem 4.1 in Shankar's paper.

```

theorem four-one:
  assumes ie1:  $\beta \leq rmin$ 
  and ie2:  $\mu \leq \delta S$ 
  and ie3:  $\gamma 1 \delta S \leq \delta S$ 
shows okmaxsync i  $\delta S$ 
proof(induct i)
  show okmaxsync 0  $\delta S$ 
  proof-
  {
    fix p q
    assume corr-p: correct p (max (te p 0) (te q 0))
    assume corr-q: correct q (max (te p 0) (te q 0))

    from corr-p synch0 have cp0: correct p 0 by simp
    from corr-q synch0 have cq0: correct q 0 by simp

    from synch0 cp0 cq0 IClock
    have IC-eq-PC:
       $|IC\ p\ 0\ (max\ (te\ p\ 0)\ (te\ q\ 0)) - IC\ q\ 0\ (max\ (te\ p\ 0)\ (te\ q\ 0))|$ 
      =  $|PC\ p\ 0 - PC\ q\ 0|$  (is ?T1 = ?T2)
      by(simp add: Adj-def)

    from ie2 init synch0 cp0 have range1:  $0 \leq PC\ p\ 0 \wedge PC\ p\ 0 \leq \delta S$ 
      by auto
    from ie2 init synch0 cq0 have range2:  $0 \leq PC\ q\ 0 \wedge PC\ q\ 0 \leq \delta S$ 
      by auto
    have ?T2  $\leq \delta S$ 
    proof cases
      assume A:  $PC\ p\ 0 < PC\ q\ 0$ 
      from A range1 range2 show ?thesis
        by(auto simp add: real-abs-def)
      next
      assume notA:  $\neg (PC\ p\ 0 < PC\ q\ 0)$ 
      from notA range1 range2 show ?thesis
        by(auto simp add: real-abs-def)
    qed
    from this IC-eq-PC have ?T1  $\leq \delta S$  by simp
  }
  thus ?thesis by (simp add: okmaxsync-def)
qed
next
fix i assume ind-hyp: okmaxsync i  $\delta S$ 
show okmaxsync (Suc i)  $\delta S$ 
proof-
  {
    fix p q
    assume corr-p: correct p (max (te p (i + 1)) (te q (i + 1)))
    assume corr-q: correct q (max (te p (i + 1)) (te q (i + 1)))
    let ?tp = te p (i + 1)
    let ?tq = te q (i + 1)
    let ?tpq = max ?tp ?tq
  }

```

```

have |IC p (i+1) ?tpq - IC q (i+1) ?tpq| ≤ δS (is ?E1 ≤ δS)
proof cases
  assume A: ?tq < ?tp
  from A corr-p have cp1: correct p (te p (i+1))
    by (simp add: max-def)
  from A corr-q have cq1: correct q (te p (i+1))
    by (simp add: max-def)
  from A
  have Eq1: ?E1 = |IC p (i+1) (te p (i+1)) - IC q (i+1) (te p (i+1))|
    (is ?E1 = ?E2)
    by (simp add: max-def)
  from A cp1 cq1 corr-p corr-q ind-hyp ie1 ie2 ie3
    four-one-ind-half
  have ?E2 ≤ δS by (simp)
  from this Eq1 show ?thesis by simp
next
  assume notA: ¬ (?tq < ?tp)
  from this corr-p have cp2: correct p (te q (i+1))
    by (simp add: max-def)
  from notA corr-q have cq2: correct q (te q (i+1))
    by (simp add: max-def)
  from notA
  have Eq2: ?E1 = |IC q (i+1) (te q (i+1)) - IC p (i+1) (te q (i+1))|
    (is ?E1 = ?E3)
    by (simp add: max-def abs-minus-commute)
  from notA have ?tp ≤ ?tq by simp
  from this cp2 cq2 ind-hyp ie1 ie2 ie3 four-one-ind-half
  have ?E3 ≤ δS
    by simp
  from this Eq2 show ?thesis by (simp)
qed
}
thus ?thesis by (simp add: okmaxsync-def)
qed
qed

```

lemma VC-cfn:

```

assumes corr-p: correct p (te p (i+1))
and ie: te p (i+1) < te p (i+2)
shows VC p (te p (i+1)) = cfn p (ϑ p (i+1))
proof-
  from ie corr-p VClock have VC p (te p (i+1)) = IC p (i+1) (te p (i+1))
    by simp
  moreover
  from corr-p IClock
  have IC p (i+1) (te p (i+1)) = PC p (te p (i+1)) + Adj p (i+1)
    by blast
  moreover
  have PC p (te p (i+1)) + Adj p (i+1) = cfn p (ϑ p (i+1))
    by(simp add: Adj-def)
  ultimately show ?thesis by simp
qed

```

Lemma for the inductive case in Theorem 4.2

lemma *four-two-ind*:
assumes *ie1*: $\beta \leq rmin$
and *ie2*: $\mu \leq \delta S$
and *ie3*: $\gamma^1 \delta S \leq \delta S$
and *ie4*: $\gamma^2 \delta S \leq \delta$
and *ie5*: $\gamma^3 \delta S \leq \delta$
and *ie6*: $te\ q\ (i+1) \leq te\ p\ (i+1)$
and *ind-hyp*: *okClocks* *p q i*
and *t-bound1*: $0 \leq t$
and *t-bound2*: $t < \max (te\ p\ (i+1)) (te\ q\ (i+1))$
and *t-bound3*: $\max (te\ p\ i) (te\ q\ i) \leq t$
and *tpq-bound*: $\max (te\ p\ i) (te\ q\ i) < \max (te\ p\ (i+1)) (te\ q\ (i+1))$
and *corr-p*: *correct* *p t*
and *corr-q*: *correct* *q t*
shows $|VC\ p\ t - VC\ q\ t| \leq \delta$
proof *cases*
assume *A*: $t < te\ q\ (i+1)$

let *?tpq* = $\max (te\ p\ i) (te\ q\ i)$

have *Eq1*: $te\ p\ i \leq t \wedge te\ q\ i \leq t$
proof *cases*
assume $te\ p\ i \leq te\ q\ i$
from *this* *t-bound3* **show** *?thesis* **by** (*simp add: max-def*)
next
assume $\neg (te\ p\ i \leq te\ q\ i)$
from *this* *t-bound3* **show** *?thesis* **by** (*simp add: max-def*)
qed

from *ie6* **have** *tp-max*: $\max (te\ p\ (i+1)) (te\ q\ (i+1)) = te\ p\ (i+1)$
by(*simp add: max-def*)
from *this* *t-bound2* **have** *Eq2*: $t < te\ p\ (i+1)$ **by** *simp*

from *VClock* *Eq1 Eq2 corr-p* **have** *Eq3*: $VC\ p\ t = IC\ p\ i\ t$ **by** *simp*

from *VClock Eq1 A corr-q* **have** *Eq4*: $VC\ q\ t = IC\ q\ i\ t$ **by** *simp*
from *Eq3 Eq4* **have** *Eq5*: $|VC\ p\ t - VC\ q\ t| = |IC\ p\ i\ t - IC\ q\ i\ t|$
by *simp*

from *t-bound3 corr-p corr-q correct-closed*
have *corr-tpq*: *correct* *p ?tpq* \wedge *correct* *q ?tpq*
by(*blast*)

from *t-bound3 IC-bd corr-p corr-q*
have *Eq6*: $|IC\ p\ i\ t - IC\ q\ i\ t| \leq |IC\ p\ i\ ?tpq - IC\ q\ i\ ?tpq|$
 $+ 2 * \rho * (t - ?tpq)$ (**is** *?E1* \leq *?E2*)
by(*blast*)

from *ie1 ie2 ie3 four-one* **have** *okmaxsync i* δS **by** *simp*

from *this corr-tpq* **have** $|IC\ p\ i\ ?tpq - IC\ q\ i\ ?tpq| \leq \delta S$
by(*simp add: okmaxsync-def*)

from *Eq6 this* **have** *Eq7: ?E1* $\leq \delta S + 2 * \rho * (t - ?tpq)$ **by** *simp*

from *corr-p Eq2 rts0* **have** $t - te\ p\ i \leq rmax$ **by** *simp*

from *this* **have** $t - ?tpq \leq rmax$ **by** (*simp add: max-def*)

from *this constants-ax* **have** $2 * \rho * (t - ?tpq) \leq 2 * \rho * rmax$

by (*simp add: real-mult-le-cancel-iff1*)

hence $\delta S + 2 * \rho * (t - ?tpq) \leq \delta S + 2 * \rho * rmax$

by *simp*

from *this Eq7* **have** *?E1* $\leq \delta S + 2 * \rho * rmax$ **by** *simp*

from *this Eq5 ie4* **show** *?thesis* **by** (*simp add: \gamma2-def*)

next

assume $\neg (t < te\ q\ (i+1))$

hence *B: te q (i+1) ≤ t* **by** *simp*

from *ie6 t-bound2*

have *tp-max: max (te p (i+1)) (te q (i+1)) = te p (i+1)*

by (*simp add: max-def*)

have $te\ p\ i \leq \max (te\ p\ i) (te\ q\ i)$

by (*simp add: max-def*)

from *this t-bound3* **have** *tp-bound1: te p i ≤ t* **by** *simp*

from *tp-max t-bound2* **have** *tp-bound2: t < te p (i+1)* **by** *simp*

have *tq-bound1: t < te q (i+2)*

proof (*rule ccontr*)

assume $\neg (t < te\ q\ (i+2))$

hence *C: te q (i+2) ≤ t* **by** *simp*

from *C corr-q correct-closed*

have *corr-q-t2: correct q (te q (i+2))* **by** *blast*

have $te\ q\ (i+1) + \beta \leq t$

proof–

from *corr-q-t2 rts1d* **have** $rmin \leq te\ q\ (i+2) - te\ q\ (i+1)$

by *simp*

from *this ie1* **have** $\beta \leq te\ q\ (i+2) - te\ q\ (i+1)$

by *simp*

hence $te\ q\ (i+1) + \beta \leq te\ q\ (i+2)$ **by** *simp*

from *this C* **show** *?thesis* **by** *simp*

qed

from *this corr-p corr-q rts2a* **have** $te\ p\ (i+1) \leq t$

by *blast*

hence $\neg (t < te\ p\ (i+1))$ **by** *simp*

from *this tp-bound2* **show** *False* **by** *simp*

qed

from *tq-bound1 B* **have** *tq-bound2: te q (i+1) < te q (i+2)* **by** *simp*

from *B tp-bound2* **have** *tq-bound3: te q (i+1) < te p (i+1)*

by *simp*

from *B corr-p correct-closed*

have *corr-p-tq1: correct p (te q (i+1))* **by** *blast*

```

from B correct-closed corr-q
have corr-q-tq1: correct q (te q (i+1)) by blast

from corr-p-tq1 corr-q-tq1 beta-bound1
have tq-bound4: te p i ≤ te q (i+1)
  by(simp add: le-diff-eq)

from tq-bound1 VClock B corr-q
have Eq1: VC q t = IC q (i+1) t by simp

from VClock tp-bound1 tp-bound2 corr-p
have Eq2: VC p t = IC p i t by simp

from Eq1 Eq2 have Eq3: |VC p t - VC q t| = |IC p i t - IC q (i+1) t|
  by simp

from B corr-p corr-q IC-bd
have  $|IC p i t - IC q (i+1) t| \leq$ 
   $|IC p i (te q (i+1)) - IC q (i+1) (te q (i+1))| + 2*\rho*(t - te q (i+1))$ 
  by simp

from this Eq3
have VC-split: |VC p t - VC q t| ≤
   $|IC p i (te q (i+1)) - IC q (i+1) (te q (i+1))| + 2*\rho*(t - te q (i+1))$ 
  by simp

from tq-bound2 VClock corr-q-tq1
have Eq4: VC q (te q (i+1)) = IC q (i+1) (te q (i+1)) by simp

from this tq-bound2 VC-cfn corr-q-tq1
have Eq5: IC q (i+1) (te q (i+1)) = cfn q (ϑ q (i+1)) by simp

hence IC-eq-cfn: IC p i (te q (i+1)) - IC q (i+1) (te q (i+1)) =
   $IC p i (te q (i+1)) - cfn q (ϑ q (i+1))$ 
  (is ?E1 = ?E2)
  by simp

let ?f = ϑ q (i+1)
let ?ppred = λ l. correct l (te q (i+1))
let ?X = 2*Λ + δS + 2*\rho*(rmax + β)

have  $\forall l m. ?ppred l \wedge ?ppred m \longrightarrow |\vartheta q (i+1) l - \vartheta q (i+1) m| \leq ?X$ 
proof-
  {
    fix l :: process
    fix m :: process
    assume corr-l: ?ppred l
    assume corr-m: ?ppred m

    let ?tlm = max (te l i) (te m i)
    have tlm-bound: ?tlm ≤ te q (i+1)
    proof-
  }

```

from *corr-l corr-q-tq1 beta-bound1* **have** $te\ l\ i \leq te\ q\ (i+1)$
by (*simp add: le-diff-eq*)
moreover
from *corr-m corr-q-tq1 beta-bound1* **have** $te\ m\ i \leq te\ q\ (i+1)$
by (*simp add: le-diff-eq*)
ultimately show *?thesis* **by** *simp*
qed

from *tlm-bound corr-l corr-m correct-closed*
have *corr-tlm: correct l ?tlm \wedge correct m ?tlm*
by *blast*

have $|IC\ l\ i\ ?tlm - IC\ m\ i\ ?tlm| \leq \delta S$
proof–
from *ie1 ie2 ie3 four-one* **have** *okmaxsync i δS*
by *simp*
from *this corr-tlm* **show** *?thesis* **by**(*simp add: okmaxsync-def*)
qed

from *this corr-l corr-m corr-q-tq1 theta-bound*
have $|\vartheta\ q\ (i+1)\ l - \vartheta\ q\ (i+1)\ m| \leq ?X$ **by** *simp*
}
thus *?thesis* **by** *blast*
qed

hence *readOK: okRead1 ($\vartheta\ q\ (i+1)$) ?X ?ppred*
by(*simp add: okRead1-def*)

let $?E3 = cfn\ q\ (\vartheta\ q\ (i+1)) - \vartheta\ q\ (i+1)\ p$
let $?E4 = \vartheta\ q\ (i+1)\ p - IC\ p\ i\ (te\ q\ (i+1))$

have $|?E2| = |?E3 + ?E4|$ **by** (*simp add: real-abs-def*)
hence *Eq8: $|?E2| \leq |?E3| + |?E4|$* **by** (*simp add: real-abs-def*)

from *correct-count* **have** *ppredOK: np - maxfaults \leq count ?ppred np*
by *simp*
from *readOK ppredOK corr-p-tq1 corr-q-tq1 acc-prsv*
have $|?E3| \leq \alpha\ ?X$
by *blast*
from *this Eq8* **have** *Eq9: $|?E2| \leq \alpha\ ?X + |?E4|$* **by** *simp*

from *corr-p-tq1 corr-q-tq1 readerror*
have $|?E4| \leq \Lambda$ **by** *simp*

from *this Eq9* **have** *Eq10: $|?E2| \leq \alpha\ ?X + \Lambda$* **by** *simp*

from *this VC-split IC-eq-cfn*
have *almost-right:*
 $|VC\ p\ t - VC\ q\ t| \leq$
 $\alpha\ ?X + \Lambda + 2*\rho*(t - te\ q\ (i+1))$
by *simp*

have $t - te\ q\ (i+1) \leq \beta$
proof (*rule ccontr*)

assume $\neg (t - te\ q\ (i+1) \leq \beta)$
hence $te\ q\ (i+1) + \beta \leq t$ **by** *simp*
from *this corr-p corr-q rts2a* **have** $te\ p\ (i+1) \leq t$
by *auto*
hence $\neg (t < te\ p\ (i+1))$ **by** *simp*
from *this tp-bound2* **show** *False*
by *simp*
qed

from *this constants-ax*
have $\alpha\ ?X + \Lambda + 2*\rho*(t - te\ q\ (i+1))$
 $\leq \alpha\ ?X + \Lambda + 2*\rho*\beta$
by (*simp*)

from *this almost-right*
have $|VC\ p\ t - VC\ q\ t| \leq \alpha\ ?X + \Lambda + 2*\rho*\beta$
by *arith*

from *this ie5* **show** *?thesis* **by** (*simp add: \gamma3-def*)
qed

Theorem 4.2 in Shankar's paper.

theorem *four-two:*

assumes *ie1: \beta \leq rmin*
and *ie2: \mu \leq \delta S*
and *ie3: \gamma1 \delta S \leq \delta S*
and *ie4: \gamma2 \delta S \leq \delta*
and *ie5: \gamma3 \delta S \leq \delta*

shows *okClocks p q i*

proof (*induct i*)

show *okClocks p q 0*

proof–

{
fix *t :: time*
assume *t-bound1: 0 \leq t*
assume *t-bound2: t < max (te p 0) (te q 0)*
assume *corr-p: correct p t*
assume *corr-q: correct q t*
from *t-bound2 synch0* **have** $t < 0$
by (*simp add: max-def*)
from *this t-bound1* **have** *False* **by** *simp*
hence $|VC\ p\ t - VC\ q\ t| \leq \delta$ **by** *simp*

}
thus *?thesis* **by** (*simp add: okClocks-def*)

qed

next

fix *i::nat* **assume** *ind-hyp: okClocks p q i*

show *okClocks p q (Suc i)*

proof –

{
fix *t :: time*
assume *t-bound1: 0 \leq t*
assume *t-bound2: t < max (te p (i+1)) (te q (i+1))*

```

assume corr-p: correct p t
assume corr-q: correct q t

let ?tpq1 = max (te p i) (te q i)
let ?tpq2 = max (te p (i+1)) (te q (i+1))

have  $|VC\ p\ t - VC\ q\ t| \leq \delta$ 
proof cases
  assume tpq-bound: ?tpq1 < ?tpq2
  show ?thesis
  proof cases
    assume  $t < ?tpq1$ 
    from t-bound1 this corr-p corr-q ind-hyp
    show ?thesis by(simp add: okClocks-def)
  next
    assume  $\neg (t < ?tpq1)$ 
    hence tpq-le-t:  $?tpq1 \leq t$  by arith

    show ?thesis
    proof cases
      assume A:  $te\ q\ (i+1) \leq te\ p\ (i+1)$ 

      from this tpq-le-t tpq-bound ie1 ie2 ie3 ie4 ie5
      ind-hyp t-bound1 t-bound2
      corr-p corr-q tpq-bound four-two-ind
      show ?thesis by(simp)
    next
      assume  $\neg (te\ q\ (i+1) \leq te\ p\ (i+1))$ 
      hence B:  $te\ p\ (i+1) \leq te\ q\ (i+1)$  by simp

      from ind-hyp okClocks-sym have ind-hyp1: okClocks q p i
      by blast

      have maxsym1:  $max\ (te\ p\ (i+1))\ (te\ q\ (i+1)) = max\ (te\ q\ (i+1))\ (te\ p\ (i+1))$ 
      by (simp add: max-def)
      have maxsym2:  $max\ (te\ p\ i)\ (te\ q\ i) = max\ (te\ q\ i)\ (te\ p\ i)$ 
      by (simp add: max-def)

      from maxsym1 t-bound2
      have t-bound21:  $t < max\ (te\ q\ (i+1))\ (te\ p\ (i+1))$ 
      by simp

      from maxsym1 maxsym2 tpq-bound
      have tpq-bound1:  $max\ (te\ q\ i)\ (te\ p\ i) < max\ (te\ q\ (i+1))\ (te\ p\ (i+1))$ 
      by simp
      from maxsym2 tpq-le-t
      have tpq-le-t1:  $max\ (te\ q\ i)\ (te\ p\ i) \leq t$  by simp

      from B tpq-le-t1 tpq-bound1 ie1 ie2 ie3 ie4 ie5
      ind-hyp1 t-bound1 t-bound21
      corr-p corr-q tpq-bound four-two-ind
      have  $|VC\ q\ t - VC\ p\ t| \leq \delta$  by(simp)
      thus ?thesis by (simp add: abs-minus-commute)

```

```

    qed
  qed
next
  assume  $\neg (?tpq1 < ?tpq2)$ 
  hence  $?tpq2 \leq ?tpq1$  by arith
  from t-bound2 this have  $t < ?tpq1$  by arith
  from t-bound1 this corr-p corr-q ind-hyp
  show ?thesis by (simp add: okClocks-def)
qed
}
thus ?thesis by (simp add: okClocks-def)
qed
qed

```

The main theorem: all correct clocks are synchronized within the bound delta.

theorem *agreement*:

```

  assumes ie1:  $\beta \leq rmin$ 
  and ie2:  $\mu \leq \delta S$ 
  and ie3:  $\gamma1 \delta S \leq \delta S$ 
  and ie4:  $\gamma2 \delta S \leq \delta$ 
  and ie5:  $\gamma3 \delta S \leq \delta$ 
  and ie6:  $0 \leq t$ 
  and cpq: correct p t  $\wedge$  correct q t
shows  $|VC\ p\ t - VC\ q\ t| \leq \delta$ 

```

proof—

```

  from ie6 cpq event-bound have  $\exists i :: nat. t < \max (te\ p\ i) (te\ q\ i)$ 
  by simp
  from this obtain i :: nat where t-bound:  $t < \max (te\ p\ i) (te\ q\ i) ..$ 

```

```

  from t-bound ie1 ie2 ie3 ie4 ie5 four-two have okClocks p q i
  by simp

```

```

  from ie6 this t-bound cpq show ?thesis
  by (simp add: okClocks-def)

```

qed

end

References

- [1] L. Lamport and P. M. Melliar-Smith. Synchronizing clocks in the presence of faults. *J. ACM*, 32(1):52–78, 1985.
- [2] J. Lundelius and N. Lynch. A new fault-tolerant algorithm for clock synchronization. In *Proceedings of PODC '84*, pages 75–88, New York, NY, USA, 1984. ACM Press.
- [3] F. B. Schneider. Understanding protocols for byzantine clock synchronization. Technical Report 87-859, Department of Computer Science, Cornell University, August 1987.

- [4] N. Shankar. Mechanical verification of a generalized protocol for byzantine fault tolerant clock synchronization. In J. Vytopil, editor, *Formal Techniques in Real-Time and Fault-Tolerant Systems*, volume 571 of *LNCS*. Springer Verlag, Jan. 1992.