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

Alwen Tiu
August 28, 2014

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

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.

\textbf{type-synonym} \ process = nat \\
\textbf{type-synonym} \ event = nat \\
\textbf{type-synonym} \ time = real \\
\textbf{type-synonym} \ 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 \text{Clocktime and} \\
VC :: [process, time] \Rightarrow \text{Clocktime and} \\
te :: [process, event] \Rightarrow \text{time and}
\[ \vartheta :: [\text{process, event}] \Rightarrow (\text{process} \Rightarrow \text{Clocktime}) \quad \text{and} \]

\[ \text{IC} :: [\text{process, event, time}] \Rightarrow \text{Clocktime} \quad \text{and} \]

\[ \text{correct} :: [\text{process, time}] \Rightarrow \text{bool} \quad \text{and} \]

\[ \text{cfn} :: [\text{process, (process} \Rightarrow \text{Clocktime)}] \Rightarrow \text{Clocktime} \quad \text{and} \]

\[ \pi :: [\text{Clocktime, Clocktime}] \Rightarrow \text{Clocktime} \quad \text{and} \]

\[ \alpha :: \text{Clocktime} \Rightarrow \text{Clocktime} \]

definition

\[ \text{count} :: [\text{process} \Rightarrow \text{bool}, \text{process}] \Rightarrow \text{nat} \quad \text{where} \]

\[ \text{count} f n = \text{card} \{ p. \ p < n \land f p \} \]

definition

\[ \text{Adj} :: [\text{process, event}] \Rightarrow \text{Clocktime} \quad \text{where} \]

\[ \text{Adj} = (\lambda p i. \text{if } 0 < i \text{ then } \text{cfn} p (\vartheta p i) - PC p (te p i) \quad \text{else } 0) \]

definition

\[ \text{okRead1} :: [\text{process} \Rightarrow \text{Clocktime, Clocktime, process} \Rightarrow \text{bool}] \Rightarrow \text{bool} \quad \text{where} \]

\[ \text{okRead1} f x \text{ppred} \longleftrightarrow (\forall l m. \text{ppred} l \land \text{ppred} m \longrightarrow |f l - f m| \leq x) \]

definition

\[ \text{okRead2} :: [\text{process} \Rightarrow \text{Clocktime, process} \Rightarrow \text{Clocktime, Clocktime, process} \Rightarrow \text{bool}] \Rightarrow \text{bool} \quad \text{where} \]

\[ \text{okRead2} f g x \text{ppred} \longleftrightarrow (\forall p. \text{ppred} p \longrightarrow |f p - g p| \leq x) \]

definition

\[ \text{rho-bound1} :: [[\text{process, time}] \Rightarrow \text{Clocktime}] \Rightarrow \text{bool} \quad \text{where} \]

\[ \text{rho-bound1} C \longleftrightarrow (\forall p s t. \text{correct} p t \land s \leq t \longrightarrow C p t - C p s \leq (t - s)*(1 + \rho)) \]

definition

\[ \text{rho-bound2} :: [[\text{process, time}] \Rightarrow \text{Clocktime}] \Rightarrow \text{bool} \quad \text{where} \]

\[ \text{rho-bound2} C \longleftrightarrow (\forall p s t. \text{correct} p t \land s \leq t \longrightarrow (t - s)*(1 - \rho) \leq C p t - C p s) \]

2.2 Clock conditions

Some general assumptions

axiomatization where

\[ \text{constants-ax: } 0 < \beta \land 0 < \mu \land 0 < r_{\text{min}} \]

\[ \land r_{\text{min}} \leq r_{\text{max}} \land 0 < \varrho \land 0 < n_p \land \text{maxfaults} \leq n_p \]

axiomatization where

\[ \text{PC-monotone: } \forall p s t. \text{correct} p t \land s \leq t \longrightarrow PC p s \leq PC p t \]

axiomatization where

\[ \text{VClock: } \forall p t i. \text{correct} p t \land te p i \leq t \land t < \text{te p} (i + 1) \longrightarrow \text{VC p t} = \text{IC p i t} \]
axiomatization where
\[ IC\Clock: \forall\ p \ t \ i. \ correct\ p\ t \rightarrow IC\ p\ i\ t = PC\ p\ t + Adj\ p\ i \]

Condition 1: initial skew

axiomatization where
\[ init: \forall\ p. \ correct\ p\ 0 \rightarrow 0 \leq PC\ p\ 0 \land PC\ p\ 0 \leq \mu \]

Condition 2: bounded drift

axiomatization where
\[ rate-1: \forall\ p\ s\ t.\ correct\ p\ t \land s \leq t \rightarrow PC\ p\ t - PC\ p\ s \leq (t - s) \cdot (t + \rho) \text{ and} \]
\[ rate-2: \forall\ p\ s\ t.\ correct\ p\ t \land s \leq t \rightarrow (t - s) \cdot (1 - \rho) \leq PC\ p\ t - PC\ p\ s \]

Condition 3: bounded interval

axiomatization where
\[ rts0: \forall\ p\ t\ i.\ correct\ p\ t \land te\ p\ (i+1) \rightarrow t - te\ p\ i \leq rmax \text{ and} \]
\[ rts1: \forall\ p\ t\ i.\ correct\ p\ t \land te\ p\ (i+1) \leq t \rightarrow rmin \leq t - te\ p\ i \]

Condition 4: bounded delay

axiomatization where
\[ rts2a: \forall\ p\ q\ t\ i.\ correct\ p\ t \land correct\ q\ t \land te\ q\ i + \beta \leq t \rightarrow te\ p\ i \leq t \text{ and} \]
\[ rts2b: \forall\ p\ q\ t\ i.\ correct\ p\ (te\ p\ i) \land correct\ q\ (te\ q\ i) \rightarrow abs(te\ p\ i - te\ q\ i) \leq \beta \]

Condition 5: initial synchronization

axiomatization where
\[ synch0: \forall\ p.\ te\ p\ 0 = 0 \]

Condition 6: nonoverlap

axiomatization where
\[ nonoverlap: \beta \leq rmin \]

Condition 7: reading errors

axiomatization where
\[ reader\ err: \forall\ p\ q\ i.\ correct\ p\ (te\ p\ (i+1)) \land correct\ q\ (te\ p\ (i+1)) \rightarrow \]
\[ abs(\vartheta\ p\ (i+1)\ q - IC\ q\ i\ (te\ p\ (i+1))) \leq \Lambda \]

Condition 8: bounded faults

axiomatization where
\[ correct\closed: \forall\ p\ s\ t.\ s \leq t \land correct\ p\ t \rightarrow correct\ p\ s \text{ and} \]
\[ correct\count: \forall\ t.\ np - maxfautls \leq count\ (\lambda\ p.\ correct\ p\ t)\ np \]

Condition 9: Translation invariance

axiomatization where
\[ trans-inv: \forall\ p\ f\ x.\ 0 \leq x \rightarrow cfn\ p\ (\lambda\ y.\ f\ y + x) = cfn\ p\ f + x \]

Condition 10: precision enhancement

axiomatization where
\[ prec-ench: \forall\ ppred\ p\ q\ f\ g\ x\ y.\]
\[ np - maxfautls \leq count\ ppred\ np \land \]
\[ okRead1 f\ y\ ppred \land okRead1 g\ y\ ppred \land \]
\[
\text{okRead2 } f \ g \ x \ \text{ppred} \land \text{ppred } p \land \text{ppred } q \\
\rightarrow \text{abs} (cfn \ p \ f - \text{cfn } q \ g) \leq \pi \ x \ y
\]

Condition 11: accuracy preservation

axiomatization where

\[
\forall \text{ppred } p \ q \ f \ x. \ \text{okRead1 } f \ x \ \text{ppred} \land \text{ppred } p \land \text{ppred } q \\
\land \text{ppred } p \land \text{ppred } q \rightarrow \text{abs} (cfn \ p \ f - \text{f } q) \leq \alpha \ x
\]

2.2.1 Some derived properties of clocks

**lemma rts0d:**
assumes \( cp: \text{correct } p \ (te \ p \ (i+1)) \)
shows \( te \ p \ (i+1) - te \ p \ i \leq \text{rmax} \)
using \( cp \) rts0 by simp

**lemma rts1d:**
assumes \( cp: \text{correct } p \ (te \ p \ (i+1)) \)
shows \( \text{rmin} \leq \text{te } p \ (i+1) - \text{te } p \ i \)
using \( cp \) rts1 by simp

**lemma rte:**
assumes \( cp: \text{correct } p \ (te \ p \ (i+1)) \)
shows \( \text{te } p \ i \leq \text{te } p \ (i+1) \)
proof–
  from \( cp \) rts1d have \( \text{rmin} \leq \text{te } p \ (i+1) - \text{te } p \ i \)
  by simp
  from this constants-ax show \( ?\text{thesis} \) by arith

qed

**lemma beta-bound1:**
assumes \( \text{corr-p: correct } p \ (te \ p \ (i+1)) \)
and \( \text{corr-q: correct } q \ (te \ p \ (i+1)) \)
shows \( 0 \leq \text{te } p \ (i+1) - \text{te } q \ i \)
proof–
  from \( \text{corr-p} \) rte have \( \text{te } p \ i \leq \text{te } p \ (i+1) \)
  by simp
  from this \( \text{corr-p} \) correct-closed have \( \text{corr-pi: correct } p \ (te \ p \ i) \)
  by blast
  from \( \text{corr-p} \) rts1d nonoverlap have \( \text{rmin} \leq \text{te } p \ (i+1) - \text{te } p \ i \)
  by simp
  from this nonoverlap have \( \beta \leq \text{te } p \ (i+1) - \text{te } p \ i \) by simp
  hence \( \text{te } p \ i + \beta \leq \text{te } p \ (i+1) \) by simp

  from this \( \text{corr-p} \) corr-q rts2a
  have \( \text{te } q \ i \leq \text{te } p \ (i+1) \)
  by blast
  thus \( ?\text{thesis} \) by simp

qed

**lemma beta-bound2:**
assumes \( \text{corr-p: correct } p \ (te \ p \ (i+1)) \)
and \( \text{corr-q: correct } q \ (te \ q \ i) \)
shows \( te p (i+1) - te q i \leq r_{\text{max}} + \beta \)

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: \( \text{abs}(te p i - te q i) \leq \beta \)
by simp

have Eq2: \( te p i - te q i \leq \beta \)

proof cases

assume \( te q i \leq te p i \)
from this Eq1 show \( \text{?thesis} \)
by (simp add: abs-if)

next

assume \( \neg (te q i \leq te p i) \)
from this Eq1 show \( \text{?thesis} \)
by (simp add: abs-if)

qed

from corr-p rts0d have \( te p (i+1) - te p i \leq r_{\text{max}} \)
by simp

from this split Eq2 show \( \text{?thesis} \) 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: abs-if)
from corr-p ie rb1 have Eq2: \( ?Bp - ?Bq \leq ?I*(1+\rho) - ?Bq \) (is \( ?E1 \leq ?E2 \))
by(simp add: rho-bound1-def)
from corr-q ie rb2 have \( ?I*(1 - \rho) \leq ?Bq \)
by(simp add: rho-bound2-def)
from this have Eq3: \(?E2 \leq ?I* (1+\varrho) - ?I* (1 - \varrho)\)
by(simp)

have Eq4: \(?I* (1+\varrho) - ?I* (1 - \varrho) = 2*\varrho*?I\)
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: abs-if)

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 \(?Bp \leq ?Bq\) 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 \textit{cp IClock} have \( IC \, p \, i \, t = PC \, p \, t + Adj \, p \, i \)
by simp
moreover
from \textit{cps IClock} have \( IC \, p \, i \, s = PC \, p \, s + Adj \, p \, i \)
by simp
moreover
from \textit{cp ie rate-1} have \( PC \, p \, t - PC \, p \, s \leq (t - s) \ast (1 + \varrho) \)
by simp
ultimately show \(?thesis\) by simp
qed

thus ?thesis by (simp add: \textit{rho-bound1-def})
qed

\textbf{lemma IC-rate2:}
\textit{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 \( ie: s \leq t \)
  from \textit{cp ie correct-closed} have \( cps: correct \, p \, s \)
  by blast
  have \( (t - s) \ast (1 - \varrho) \leq IC \, p \, i \, t - IC \, p \, i \, s \)
  proof
    from \textit{cp IClock} have \( IC \, p \, i \, t = PC \, p \, t + Adj \, p \, i \)
    by simp
    moreover
    from \textit{cps IClock} have \( IC \, p \, i \, s = PC \, p \, s + Adj \, p \, i \)
    by simp
    moreover
    from \textit{cp ie rate-2} have \( (t - s) \ast (1 - \varrho) \leq PC \, p \, t - PC \, p \, s \)
    by simp
    ultimately show ?thesis by simp
  qed
\}
thus ?thesis by (simp add: \textit{rho-bound2-def})
qed

Auxiliary function \textit{ICf}: we introduce this to avoid some unification problem in some tactic of isabelle.

\textbf{definition}
\textit{ICf} :: \( nat \Rightarrow (process \Rightarrow time \Rightarrow Clocktime) \) where
\( ICf \, i = (\lambda \, p \, t. \, IC \, p \, i \, t) \)

\textbf{lemma IC-bd:}
assumes \( ie: s \leq t \)
and \( corr-p: correct \, p \, t \)
and \( corr-q: correct \, q \, t \)
shows \( |IC \, p \, i \, t - IC \, q \, j \, t| \leq |IC \, p \, i \, s - IC \, q \, j \, s| + 2 \ast \varrho \ast (t - s) \)
proof
- let \(?C = ICf \, i\)
let \( \delta = ICf \, j \)
let \( \gamma = |C \, p \, t - \delta \, q \, t| \leq |C \, p \, s - \delta \, q \, s| + 2 \rho \ast (t - s) \)

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

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

from ie rb1 rb2 corr-p corr-q bounded-drift
have \( \gamma \) by simp
from this show \( \delta \) 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: \( \forall \ (i :: nat). \ correct \ (te \, p \, i) \)
      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: \( \forall \ (i :: nat). \ real \, i \ast rmin \leq te \, p \, i \)
      proof
        fix \( i :: nat \)
        show real \( i \ast rmin \leq te \, p \, i \)
        proof (induct \( i \))
          from synch0 show real \( (0::nat) \ast rmin \leq te \, p \, 0 \) by simp
        next
          fix \( i :: nat \) assume ind-hyp: \( real \, i \ast rmin \leq te \, p \, i \)
          show \( real \, (Suc \, i) \ast rmin \leq te \, p \, (Suc \, i) \)
        qed
      qed

5
proof -

have Eq1: real i * rmin + rmin = (real i + 1)*rmin
  by (simp add: distrib-right)
have Eq2: real i + 1 = real (i+1) by simp
from Eq1 Eq2
have Eq3: real i * rmin + rmin = 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 ≤ te p (i+1) − te p i
  by simp
hence Eq4: te p i + rmin ≤ te p (i+1) by simp
from ind-hyp have real i * rmin + rmin ≤ te p i + rmin
  by (simp)
from this Eq4 have real i * rmin + rmin ≤ te p (i+1)
  by simp
from this Eq3 show ?thesis by simp
qed
qed

have F4: ∀ (i::nat). real i * rmin ≤ t
proof
  fix i::nat
  from F1 have te p i ≤ t by simp
  moreover
  from F3 have real i * rmin ≤ te p i by simp
  ultimately show real i * rmin ≤ t by simp
qed

from constants-ax have 0 < rmin by simp

from this reals-Archimedean3
have Archi: ∃ (k::nat). t < real k * rmin
  by blast

from Archi obtain k::nat where C: t < real k * rmin ..

from F4 have real k * rmin ≤ t by simp
hence notC: ¬ (t < real k * rmin) by simp

from C notC show False by simp
qed
qed

2.3 Agreement property

definition γ1 x = π (2*ϱ*β + 2*Λ) (2*Λ + x + 2*ϱ*(rmax + β))
definition γ2 x = x + 2*ϱ*rmax
\[
\gamma x = \alpha (2\Lambda + x + 2\gamma \text{max}(\beta)) + \Lambda + 2\gamma \beta
\]

**Definition**

\( \text{okmaxsync} \:: \text{[nat, Clocktime]} \Rightarrow \text{bool} \)**

\[\text{okmaxsync } i \ x \leftrightarrow (\forall \ p \ q. \ \text{correct } p (\text{max} (\text{te } p \ i) (\text{te } q \ i)) \land \text{correct } q (\text{max} (\text{te } p \ i) (\text{te } q \ i)) \implies |\text{IC } p \ i (\text{max} (\text{te } p \ i) (\text{te } q \ i)) - \text{IC } q \ i (\text{max} (\text{te } p \ i) (\text{te } q \ i))| \leq x)\]

**Definition**

\( \text{okClocks} \:: \text{[process, process, nat]} \Rightarrow \text{bool} \)**

\[\text{okClocks } p \ q \ i \leftrightarrow (\forall \ t. \ 0 \leq t \land t < \text{max} (\text{te } p \ i) (\text{te } q \ i) \land \text{correct } p t \land \text{correct } q t \implies |\text{VC } p \ t - \text{VC } q \ t| \leq \delta)\]

**Lemma** \(\text{okClocks-sym} \): 

**Assumes** \(\text{ok-pq} \): \(\text{okClocks } p \ q \ i\)

**Shows** \(\text{okClocks } q \ p \ i\)

**Proof**

\[
\begin{align*}
\text{fix } t :: \text{time} \\
\text{assume } \text{ie1}: 0 \leq t \\
\text{assume } \text{ie2}: t < \text{max} (\text{te } q \ i) (\text{te } p \ i) \\
\text{assume } \text{corr-q}: \text{correct } q \ t \\
\text{assume } \text{corr-p}: \text{correct } p \ t \\
\text{have } \text{max} (\text{te } q \ i) (\text{te } p \ i) = \text{max} (\text{te } p \ i) (\text{te } q \ i) \quad \text{by } (\text{simp add: max-def}) \\
\text{from this ok-pq ie1 ie2 corr-p corr-q} \\
\text{have } |\text{VC } q \ t - \text{VC } p \ t| \leq \delta \\
\text{by} (\text{simp add: abs-minus-commute okClocks-def})
\end{align*}
\]

**Thus** \(?\text{thesis by} (\text{simp add: \text{okClocks-def})}\)

**QED**

**Lemma** \(\text{ICp-Suc} \):

**Assumes** \(\text{corr-p}: \text{correct } p (\text{te } p \ (i+1))\)

**Shows** \(\text{IC } p \ (i+1) (\text{te } p \ (i+1)) = \text{cfn } p (\emptyset p (i+1))\)

**Using** \(\text{corr-p IClock by} (\text{simp add: Adj-def})\)

**Lemma** \(\text{IC-trans-inv} \):

**Assumes** \(\text{ie1}: \text{te } q (i+1) \leq \text{te } p (i+1)\)

**And** \(\text{corr-p}: \text{correct } p (\text{te } p (i+1))\)

**And** \(\text{corr-q}: \text{correct } q (\text{te } p (i+1))\)

**Shows**

\(\text{IC } q (i+1) (\text{te } p (i+1)) = \text{cfn } q (\lambda n. \emptyset q (i+1) n \oplus (\text{PC } q (\text{te } p (i+1)) - \text{PC } q (\text{te } q (i+1))))\)

(IS \(?T1 = ?T2)\)

**Proof**

\[
\begin{align*}
\text{let } ?X = \text{PC } q (\text{te } p (i+1)) - \text{PC } q (\text{te } q (i+1)) \\
\text{from corr-q ie1 PC-monotone have posX: } 0 \leq ?X \\
\text{by} (\text{simp add: le-diff-eq})
\end{align*}
\]
from \texttt{IClock corr-q} have \(?T1 = \text{cfn} q (\emptyset q (i+1)) + ?X\) 
by(simp add: Adj-def)

from this \texttt{posX} trans-inv show \(?\text{thesis by simp}\)
qed

lemma beta-rho:
assumes \texttt{ie: te q (i+1) \leq te p (i+1)}
and \texttt{corr-p: correct p (te p (i+1))}
and \texttt{corr-q: correct q (te p (i+1))}
and \texttt{corr-l: correct l (te p (i+1))}
shows \(|PC l (te p (i+1)) - PC l (te q (i+1))| - (te p (i+1) - te q (i+1))| \leq \beta * q\)
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 \texttt{ie} have \texttt{posD: 0 \leq ?D by simp}

from \texttt{ie PC-monotone corr-l} have \texttt{posX: 0 \leq ?X by (simp add: le-diff-eq)}
from \texttt{ie corr-l rate-1} have \texttt{bound1: ?X \leq ?D * (1 + q) by simp}
from \texttt{ie corr-l correct-closed have corr-l-tq: correct l (te q (i+1)) by blast}
from \texttt{ie corr-l-tq corr-p rts2b} have \texttt{corr-l-tq: correct q (te q (i+1)) by blast}
from this \texttt{constants-ax posD} have D-beta: \(?D * q \leq \beta * q\) by (simp add: abs-if)

show \(?\text{thesis}\)
proof cases
  assume \(A: ?D \leq \beta\)
  from this \texttt{have absEq: \(\|X - ?D\| = \|X - ?D\|\)}
  by(simp add: abs-if)
from \texttt{bound1} have \texttt{bound2: ?X - ?D \leq ?D * q by simp}
  by(simp add: mult.commute distrib-right)
from \texttt{D-beta absEq bound2} show \(?\text{thesis by simp}\)
next
  assume \(\neg A: \neg (\beta \leq \beta)\)
from this \texttt{have absEq2: \(\|X - ?D\| = \|D - ?X\|\)}
  by(simp add: abs-if)
from \texttt{ie corr-l rate-2} have \texttt{bound3: ?D * (1 - q) \leq ?X by simp}
from this \texttt{have ?D - ?X \leq \beta * q by (simp add: algebra-simps)}
from this \texttt{absEq2 D-beta} show \(?\text{thesis by simp}\)
qed
qed

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

lemma pe-cond1:
assumes \texttt{ie: te q (i+1) \leq te p (i+1)}
and \texttt{corr-p: correct p (te p (i+1))}
and \texttt{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: abs-if)

hence Split: \(?M \leq |?Xq - ?D| + |?T - ?D|\)
by(simp add: abs-if)

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: abs-if)

have Eq2: \(|?E2| \leq |?T - ?ICT| + |?ICT - ?D|\)
by(simp add: abs-if)

have Eq3: \(|?T - ?ICT| \leq |?RE1| + |?RE2|\)
by(simp add: abs-if)

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 \( t e \ p \ i \ \leq \ t e \ p \ (i+1) \)
  by (simp)
from this corr-l correct-closed have corr-l-tpi: correct l \( (t e \ p \ i) \)
  by blast
from corr-q-tq rte have \( t e \ q \ i \ \leq \ t e \ q \ (i+1) \)
  by simp
from this corr-l-tq correct-closed have corr-l-tqi: correct l \( (te q i) \)
  by blast
d from IClock corr-l
have \( F1: IC l i \ (t e \ p \ (i+1)) = PC l \ (t e \ p \ (i+1)) + Adj l i \)
  by (simp)
d from IClock corr-l-tq
have \( F2: IC l i \ (t e \ q \ (i+1)) = PC l \ (t e \ q \ (i+1)) + Adj l i \)
  by simp
d from \( F1 \ F2 \) show \(?\thesis\) by (simp)
qed

d from this XID have Eq8: \( |ICT - ?D| \leq \beta * \varrho \)
  by arith
d from Eq1 Eq2 Eq6 Eq8 show \(?\thesis\)
  by (simp)
qed

d from Split XqD TD have F1: \( ?M \leq 2 * \beta * \varrho + 2 * \Lambda \)
  by (simp)
d have \( F2: 2 * \varrho * \beta + 2 * \Lambda = 2 * \beta * \varrho + 2 * \Lambda \)
  by simp
d from \( F1 \) show \(?\thesis\) by (simp only: F2)
qed

lemma pe-cond2:
  assumes ic: \( te \ m i \ \leq \ t e \ l i \)
  and corr-k: correct k \( (t e \ k \ (i+1)) \)
  and corr-l-tk: correct l \( (t e \ k \ (i+1)) \)
  and corr-m-tk: correct m \( (t e \ k \ (i+1)) \)
  and ind-hyp: \( |IC l i \ (t e \ l i) - IC m i \ (t e \ l i)| \leq \delta S \)
shows \( |\vartheta k \ (i+1) \ l - \vartheta k \ (i+1) \ m| \leq 2 * \Lambda + \delta S + 2 * \varrho * (\text{rmax} + \beta) \)
proof-
  let \( ?X = \vartheta k \ (i+1) \ l - \vartheta k \ (i+1) \ m \)
  let \( ?N = 2 * \Lambda + \delta S + 2 * \varrho * (\text{rmax} + \beta) \)
  let \( ?D1 = \vartheta k \ (i+1) \ l - IC l i \ (t e \ k \ (i+1)) \)
  let \( ?D2 = \vartheta k \ (i+1) \ m - IC m i \ (t e \ k \ (i+1)) \)
  let \( ?ICS = IC l i \ (t e \ k \ (i+1)) - IC m i \ (t e \ k \ (i+1)) \)
  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: abs-if)
  have Eq2: \(?E2 \leq |?D1 - ?D2| + |?ICS| \) by (simp add: abs-if)
from corr-l-tk corr-k beta-bound1 have ie-lk: \( te \ l \ i \leq te \ k \ (i+1) \)
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: \(|ICS| \leq |IC| + 2*\rho*(te k (i+1) - ?tlm)\)
by simp
from this ind-hyp have Eq4: \(|ICS| \leq \delta S + 2*\rho*(te k (i+1) - ?tlm)\)
by simp

from corr-l corr-k beta-bound2 have \( te k \ (i+1) - ?tlm \leq r_{\text{max}} + \beta \)
by simp
from this constants-ax have \( 2*\rho*(te k (i+1) - ?tlm) \leq 2*\rho*(r_{\text{max}} + \beta) \)
by (simp add: real-mult-le-cancel-iff2)
from this Eq4 have Eq4a: \(|ICS| \leq \delta S + 2*\rho*(r_{\text{max}} + \beta)\)
by (simp)

from corr-k corr-l-tk readerror
have Eq5: \(|D_1| \leq \Lambda \) by simp
from corr-k corr-m-tk readerror
have Eq6: \(|D_2| \leq \Lambda \) by simp

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:
\[ |IC \ l \ (max (te \ l \ i) \ (te \ m \ i)) - IC \ m \ (max (te \ l \ i) \ (te \ m \ i))| \leq \delta S \]
shows \[ |\vartheta \ p \ (i+1) \ l - \vartheta \ p \ (i+1) \ m| \leq 2*\Lambda + \delta S + 2*\rho*(r_{\text{max}} + \beta) \]
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 \(?tlm = max (te \ l \ i) \ (te \ m \ i)\)
from tli-le-tp tmi-le-tp have tml-le-tp: \(?tlm \leq te \ p \ (i+1)\)
by simp

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

let $?Y = 2*Λ + δS + 2*ϱ*(rmax + β)
show $|ϑ p (i+1) l − ϑ p (i+1) m| ≤ $?Y

proof cases
  assume $A: te m i < te l i$
  from this IC-bound
  have $|IC l i (te l i) − IC m i (te l i)| ≤ δS$
  by(simp add: max-def)
  from this $A$ corr-p corr-l corr-m pe-cond2
  show $?thesis$ by(simp)

next
  assume $¬ (te m i < te l i)$
  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 $|IC m i (te m i) − IC l i (te m i)| ≤ δS$
  by (simp add: abs-minus-commute)
  from this Eq1 corr-p corr-l corr-m pe-cond2
  have $|ϑ p (i+1) m − ϑ p (i+1) l| ≤ $?Y
  by(simp)
  thus $?thesis$ by (simp add: abs-minus-commute)
qed

lemma four-one-ind-half:
  assumes ie1: $β ≤ rmin$
  and ie2: $μ ≤ δS$
  and ie3: $γ1 δS ≤ δS$
  and ind-hyp: okmaxsync i $δS$
  and ie4: $te q (i+1) ≤ te p (i+1)$
  and corr-p: correct p $(te p (i+1))$
  and corr-q: correct q $(te p (i+1))$
  shows $|IC p (i+1) (te p (i+1)) − IC q (i+1) (te p (i+1))| ≤ δS$
proof
  let $?tpq = te p (i+1)$

  let $?f = λ n. ϑ q (i+1) n + (PC q (te p (i+1)) − PC q (te q (i+1)))$
  let $?g = ϑ p (i+1)$

  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: abs-if)
qed

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

let ?X = 2*\rho*\beta + 2*\Lambda
have \forall \ l. ?ppred \ l \to |?f \ l - ?g \ l| \leq ?X
proof -
  { fix \ l
    assume ?ppred \ l
    from ie4 corr-p corr-q this pe-cond1
    have |?f \ l - ?g \ l| \leq (2*\rho*\beta + 2*\Lambda)
    by (auto)
  }
thus ?thesis by blast
qed

hence cond1: okRead2 \ ?f \ ?g \ ?X ?ppred
  by(simp add: okRead2-def)

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

have \forall \ l m. ?ppred \ l \land ?ppred \ m \to |?f \ l - ?f \ 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 \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 ?tlm = max (te l i) (te m i)

    from tli-le-tp tmi-le-tp have tlm-le-tp: ?tlm \leq 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| \leq \delta S
    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: abs-if)

from EqAbs1 corr-q-tq corr-l-tq corr-m-tq theta-bound
have |Θ q (i+1) l − Θ q (i+1) m| ≤ ?Y
  by simp
from this EqAbs3 have |?f l − ?f m| ≤ ?Y
  by simp

} thus ?thesis by simp
qed

hence cond2a: okRead1 ?f ?Y ?ppred (simp add: okRead1-def)

have ∀ l m. ?ppred l ∧ ?ppred m −→ |?g l − ?g 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 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| ≤ ?Y by simp
  } thus ?thesis by simp
qed

hence cond2b: okRead1 ?g ?Y ?ppred (simp add: okRead1-def)

from correct-count have np − maxfaults ≤ count ?ppred np
  by simp
from this corr-p corr-q cond1 cond2a cond2b prec-enh
have |cfn q ?f − cfnp p ?g| ≤ π ?X ?Y
  by blast

from ie3 this have |cfn q ?f − cfnp p ?g| ≤ δS
  by (simp add: γ1-def)
from this Eq-IC-cfn show \( \text{thesis} \) by (simp)

qed

Theorem 4.1 in Shankar’s paper.

**theorem four-one:**
assumes \( ie1: \beta \leq r_{min} \)
and \( ie2: \mu \leq \delta S \)
and \( ie3: \gamma \delta S \leq \delta S \)
shows \( \text{okmaxsync} \ i \delta S \)

**proof** (induct \( i \))

show \( \text{okmaxsync} \ 0 \delta S \)

proof

{  
  fix \( p q \)
  assume \( \text{corr-p}: \text{correct} \ p \ (\max \ (te \ p \ 0) \ (te \ q \ 0)) \)
  assume \( \text{corr-q}: \text{correct} \ q \ (\max \ (te \ p \ 0) \ (te \ q \ 0)) \)

  from \( \text{corr-p} \ \text{synch0} \) have \( cp0: \text{correct} \ p \ 0 \) by simp
  from \( \text{corr-q} \ \text{synch0} \) have \( cq0: \text{correct} \ q \ 0 \) by simp

  from \( \text{synch0} \ cp0 \ cq0 \ \text{IClock} \) have \( \text{IC-eq-PC} \):
  \[
  |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| \ (\text{is } ?T1 = ?T2) \)
  by (simp add: Adj-def)

  from \( ie2 \ \text{init} \ \text{synch0} \ cp0 \) have \( \text{range1: } 0 \leq PC \ p \ 0 \land PC \ p \ 0 \leq \delta S \)
  by auto
  from \( ie2 \ \text{init} \ \text{synch0} \ cq0 \) have \( \text{range2: } 0 \leq PC \ q \ 0 \land PC \ q \ 0 \leq \delta S \)
  by auto

  have \( ?T2 \leq \delta S \)
  proof cases
    assume \( A: PC \ p \ 0 < PC \ q \ 0 \)
    from \( A \ \text{range1 range2} \) show \( \text{thesis} \)
    by (auto simp add: abs-if)
  next
    assume \( \text{notA: } \neg \ (PC \ p \ 0 < PC \ q \ 0) \)
    from \( \text{notA range1 range2} \) show \( \text{thesis} \)
    by (auto simp add: abs-if)
  qed

  from \( \text{this IC-eq-PC} \) have \( ?T1 \leq \delta S \) by simp

  thus \( \text{thesis} \) by (simp add: okmaxsync-def)

  qed

next

fix \( i \) assume \( \text{ind-hyp: okmaxsync} \ i \delta S \)

show \( \text{okmaxsync} \ (Suc \ i) \delta S \)

proof

{  
  fix \( p q \)
  assume \( \text{corr-p}: \text{correct} \ p \ (\max \ (te \ p \ (i + 1)) \ (te \ q \ (i + 1))) \)
  assume \( \text{corr-q}: \text{correct} \ q \ (\max \ (te \ p \ (i + 1)) \ (te \ q \ (i + 1))) \)

  from \( this \ \text{Eq-IC-cfn} \) show \( \text{thesis} \) by (simp)

  qed

next


let \(?tp = te p (i + 1)\)
let \(?tq = te q (i + 1)\)
let \(?tpq = \max \(?tp \?tq\)\)

have \(|IC p (i+1) \?tpq - IC q (i+1) \?tpq| \leq \delta S\) (is \(?E1 \leq \delta S\))

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)

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 \(?tp \leq \?tq\) by simp
from this cp2 cq2 ind-hyp ie1 ie2 ie3 four-one-ind-half
have \(?E3 \leq \delta S\)
  by simp
from this Eq2 show \(?thesis\) by (simp)
qed

} thus \(?thesis\) by (simp add: okmaxsync-def)
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 \((te p (i+1)) = cfn p (\emptyset p (i+1))\)
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 \((te p (i+1)) + Adj p (i+1) = cfn p (\emptyset p (i+1))\)
ultimately show \( \text{thesis} \) by simp

\textbf{Lemma} for the inductive case in Theorem 4.2

\texttt{lemma} four-two-ind:
\begin{itemize}
  \item \texttt{assumes} ie1: \( \beta \leq \text{rmin} \)
  \item \texttt{and} ie2: \( \mu \leq \delta S \)
  \item \texttt{and} ie3: \( \delta S \leq \delta S \)
  \item \texttt{and} ie4: \( \delta S \leq \delta \)
  \item \texttt{and} ie5: \( \delta S \leq \delta \)
  \item \texttt{and} ie6: \( \text{te} \ q \ (i+1) \leq \text{te} \ p \ (i+1) \)
  \item \texttt{and} \( \text{ind-hyp: okClocks} \ p \ q \ i \)
  \item \texttt{and} \( \text{t-bound1:} \ 0 \leq t \)
  \item \texttt{and} \( \text{bound2:} \ t < \max \ (\text{te} \ p \ (i+1)) \ (\text{te} \ q \ (i+1)) \)
  \item \texttt{and} \( \text{bound3:} \ \max \ (\text{te} \ p \ i) \ (\text{te} \ q \ i) \leq t \)
  \item \texttt{and} \( \text{tpq-bound:} \ \max \ (\text{te} \ p \ i) \ (\text{te} \ q \ i) < \max \ (\text{te} \ p \ (i+1)) \ (\text{te} \ q \ (i+1)) \)
  \item \( \text{corr-p: correct } p \ t \)
  \item \( \text{corr-q: correct } q \ t \)
\end{itemize}

shows \( |V C \ p \ t - V C \ q \ t| \leq \delta \)

\texttt{proof}
\texttt{cases}
\begin{itemize}
  \item \texttt{assume} \( A: \ t < \text{te} \ q \ (i+1) \)
\end{itemize}

\texttt{let} \( \text{?tpq} = \max \ (\text{te} \ p \ i) \ (\text{te} \ q \ i) \)

\texttt{have} \( \text{Eq1:} \ \text{te} \ p \ i \leq t \ \land \ \text{te} \ q \ i \leq t \)

\texttt{proof}
\texttt{cases}
\begin{itemize}
  \item \texttt{assume} \( \text{te} \ p \ i \leq \text{te} \ q \ i \)
  \texttt{from} \( \text{this} \ \text{bound3} \ \text{show} \ \text{thesis} \) by (simp add: max-def)
\end{itemize}

\texttt{next}
\begin{itemize}
  \item \texttt{assume} \( \neg \ (\text{te} \ p \ i \leq \text{te} \ q \ i) \)
  \texttt{from} \( \text{this} \ \text{bound3} \ \text{show} \ \text{thesis} \) by (simp add: max-def)
\end{itemize}

\texttt{qed}

\texttt{from} \( \text{ie6} \ \text{have} \ \text{tp-max:} \ \max \ (\text{te} \ p \ (i+1)) \ (\text{te} \ q \ (i+1)) = \text{te} \ p \ (i+1) \)

\texttt{by(simp add: max-def)}

\texttt{from} \( \text{this} \ \text{bound2} \ \text{have} \ \text{Eq2:} \ t < \text{te} \ p \ (i+1) \) by simp

\texttt{from} \( \text{VClock} \ \text{Eq1} \ \text{Eq2} \ \text{corr-p} \ \text{have} \ \text{Eq3:} \ V C \ p \ t = I C \ p \ i \ t \) by simp

\texttt{from} \( \text{VClock} \ \text{Eq1} \ A \ \text{corr-q} \ \text{have} \ \text{Eq4:} \ V C \ q \ t = I C \ q \ i \ t \) by simp

\texttt{from} \( \text{Eq3} \ \text{Eq4} \ \text{have} \ \text{Eq5:} \ |V C \ p \ t - V C \ q \ t| = |I C \ p \ i \ t - I C \ q \ i \ t| \)

\texttt{by simp}

\texttt{from} \( \text{t-bound3} \ \text{corr-p} \ \text{corr-q} \ \text{correct-closed} \)
\texttt{have} \( \text{corr-tpq: correct } p \ ?tpq \land \text{correct } q \ ?tpq \)

\texttt{by(blast)}

\texttt{from} \( \text{t-bound3} \ \text{IC-bd} \ \text{corr-p} \ \text{corr-q} \)
\texttt{have} \( \text{Eq6:} \ |I C \ p \ i \ t - I C \ q \ i \ t| \leq |I C \ p \ i \ ?tpq - I C \ q \ i \ ?tpq| + 2 \ast q \ast (t - ?tpq) (\text{is } \ ?E1 \leq \ ?E2) \)

\texttt{by(blast)}

\texttt{21}
from \texttt{ie1 ie2 ie3 four-one} have \texttt{okmaxsync i \delta S by simp}

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

from \texttt{Eq6 this} have \texttt{Eq7: ?E1 \leq \delta S + 2*\gamma*(t - ?tpq) by simp}

from \texttt{corr-p Eq2 rts0} have \(t - tc\ p\ i \leq rmax\) by \texttt{simp}
from \texttt{this} have \(t - ?tpq \leq rmax\) by (\texttt{simp add: max-def})
from \texttt{this constants-ax} have \(2*\gamma*(t - ?tpq) \leq 2*\gamma*rmax
by (\texttt{simp add: real-mult-le-cancel-iff1})
hence \(\delta S + 2*\gamma*(t - ?tpq) \leq \delta S + 2*\gamma*rmax
by \texttt{simp}
from \texttt{this Eq7} have \texttt{?E1 \leq \delta S + 2*\gamma*rmax by simp}
from \texttt{this Eq5 ie4 show \texttt{\gamma2-def} by simp}

next
assume \(\neg(t < te\ q\ (i+1))\)
hence \texttt{B: te\ q\ (i+1) \leq t by simp}

from \texttt{ie6 t-bound2}
have \texttt{tp-max; max (te\ p\ (i+1)) (te\ q\ (i+1)) = te\ p\ (i+1)
by(\texttt{simp add: max-def})

have \(te\ p\ i \leq max\ (te\ p\ i) (te\ q\ i)
by(\texttt{simp add: max-def})

from \texttt{this t-bound3} have \texttt{tp-bound1: te\ p\ i \leq t by simp}

from \texttt{tp-max t-bound2} have \texttt{tp-bound2: t < te\ p\ (i+1) by simp}

have \texttt{tq-bound1: t < te\ q\ (i+2)
proof (rule ccontr)
assume \(\neg(t < te\ q\ (i+2))\)
hence \texttt{C: te\ q\ (i+2) \leq t by simp}

from \texttt{C corr-q correct-closed}
have \texttt{corr-q-t2: correct\ q\ (te\ q\ (i+2)) by blast}

have \(te\ q\ (i+1) + \beta \leq t
proof
from \texttt{corr-q-t2 rts1d} have \(rmin \leq te\ q\ (i+2) - te\ q\ (i+1)
by \texttt{simp}
from \texttt{this ie1} have \(\beta \leq te\ q\ (i+2) - te\ q\ (i+1)
by \texttt{simp}
hence \(te\ q\ (i+1) + \beta \leq te\ q\ (i+2) by \texttt{simp}
from \texttt{this C} show \texttt{\gamma2-def by simp}
qed
from \texttt{this corr-p corr-q rts2a} have \(te\ p\ (i+1) \leq t
by \texttt{blast}
hence \(\neg(t < te\ p\ (i+1)) by \texttt{simp}
from \texttt{this tp-bound2} show \texttt{False by simp}
qed

22
from tq-bound1 B have tq-bound2: \( \text{te } q (i+1) < \text{te } q (i+2) \) by simp
from B tp-bound2 have tq-bound3: \( \text{te } q (i+1) < \text{te } p (i+1) \)
  by simp
from B corr-p correct-closed have corr-p-tq1: correct \( p \) \( \text{te } q (i+1) \) by blast
from corr-p-tq1 corr-q correct-closed have corr-p-tq1: correct \( p \) \( \text{te } q (i+1) \) by blast
from corr-p-tq1 corr-q-tq1 beta-bound1 have tq-bound4: \( \text{te } p i \leq \text{te } q (i+1) \)
  by (simp add: le-diff-eq)
from tq-bound1 VClock B corr-q have Eq1: \( \text{VC } q \text{ t } = \text{IC } q (i+1) \text{ t} \) by simp
from VClock tp-bound1 tp-bound2 corr-p have Eq2: \( \text{VC } p \text{ t } = \text{IC } p \text{ i } \text{ t} \) by simp
from Eq1 Eq2 have Eq3: \( |\text{VC } p \text{ t } - \text{VC } q \text{ t }| = |\text{IC } p \text{ i } \text{ t } - \text{IC } q (i+1) \text{ t}| \)
  by simp
from B corr-p corr-q IC-bd have \( |\text{IC } p \text{ i } \text{ t } - \text{IC } q (i+1) \text{ t}| \leq |\text{IC } p \text{ i } \text{ t } - \text{IC } q (i+1) \text{ t}| + 2*\rho*(t - \text{te } q (i+1)) \)
  by simp
from this Eq3 have \( |\text{VC } p \text{ t } - \text{VC } q \text{ t }| \leq |\text{IC } p \text{ i } \text{ t } - \text{IC } q (i+1) \text{ t}| + 2*\rho*(t - \text{te } q (i+1)) \)
  by simp
from tq-bound2 VClock corr-q-tq1 have Eq4: \( \text{VC } q \text{ (te } q (i+1)) = \text{IC } q (i+1) \text{ (te } q (i+1)) \) by simp
from this tq-bound2 VClock-cfn corr-q-tq1 have Eq5: \( \text{IC } q (i+1) \text{ (te } q (i+1)) = \text{cfn } q (\theta q (i+1)) \) by simp
hence \( \text{IC-eq-cfn: IC } p \text{ i } \text{ t } - \text{IC } q (i+1) \text{ (te } q (i+1)) = IC } p \text{ i } \text{ t } - \text{cfn } q (\theta q (i+1)) \)
(is \( ?E1 = ?E2 \))
  by simp
let \( ?f = \theta q (i+1) \)
let \( ?\text{ppred} = \lambda l. \text{correct } l \text{ (te } q (i+1)) \)
let \( ?X = 2*\Lambda + \delta S + 2*\rho*(r_{max} + \beta) \)
have \( \forall l m. ?\text{ppred} \text{ l } \land ?\text{ppred} \text{ m } \longrightarrow |\theta q (i+1) \text{ l } - \theta q (i+1) \text{ m}| \leq ?X \)
proof -
  { fix l :: process
    fix m :: process
    assume corr-l: ?\text{ppred} \text{ 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 ≤ te q (i+1)
  by (simp add: le-diff-eq)
moreover
from corr-m corr-q-tq1 beta-bound1 have te m i ≤ 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 ∧ correct m ?tlm
  by blast

have |IC l i ?tlm − IC m i ?tlm| ≤ δS
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 |ϑ q (i+1) l − ϑ q (i+1) m| ≤ ?X by simp
\[ \alpha \ ?X + \Lambda + 2\rho \ast (t - te\ q\ (i+1)) \]

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 \( \alpha \ ?X + \Lambda + 2\rho \ast (t - te\ q\ (i+1)) \leq \alpha \ ?X + \Lambda + 2\rho \ast \beta \)
by (simp)

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

from this ie5 show \( \neg\text{thesis} \) by (simp add: \gamma3-def)

qed

Theorem 4.2 in Shankar’s paper.

defines four-two:

defines ie1: \( \beta \leq rmin \)

and ie2: \( \mu \leq \delta S \)

and ie3: \( \gamma1 \delta S \leq \delta S \)

and ie4: \( \gamma2 \delta S \leq \gamma \)

and ie5: \( \gamma3 \delta S \leq \delta \)

shows okClocks\ p\ q\ i

proof (induct i)

show okClocks\ p\ q\ 0

proof{ }
  fix \( t :: \text{time} \)
  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 \( \neg\text{thesis} \) by (simp add: okClocks-def)

qed

next

fix \( i :: \text{nat} \) assume ind-hyp: okClocks\ p\ q\ i

show okClocks\ p\ q\ (Suc\ i)
proof -
{
  fix t :: time
  assume t-bound1: 0 ≤ 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| ≤ δ
  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 ¬(t < ?tpq1)
    hence tpq-le-t: ?tpq1 ≤ t by arith
    show ?thesis
    proof cases
      assume A: te q (i+1) ≤ 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 ¬(te q (i+1) ≤ te p (i+1))
    hence B: te p (i+1) ≤ 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) ≤ t by simp

26
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| ≤ δ by (simp)
thus ?thesis by (simp add: abs-minus-commute)
qed
next
  assume ¬ (?tpq1 < ?tpq2)
hence ?tpq2 ≤ ?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: β ≤ rmin
  and ie2: μ ≤ δS
  and ie3: γ1 δS ≤ δS
  and ie4: γ2 δS ≤ δ
  and ie5: γ3 δS ≤ δ
  and ie6: 0 ≤ t
  and cpq: correct p t ∧ correct q t
shows |VC p t − VC q t| ≤ δ
proof –
  from ie6 cpq event-bound have ∃ 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