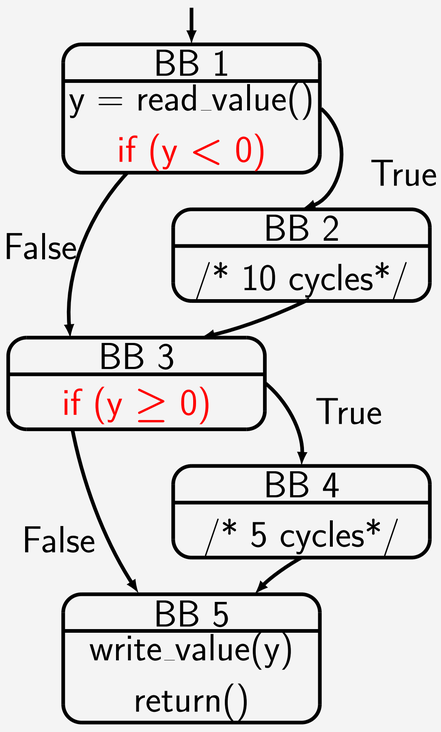
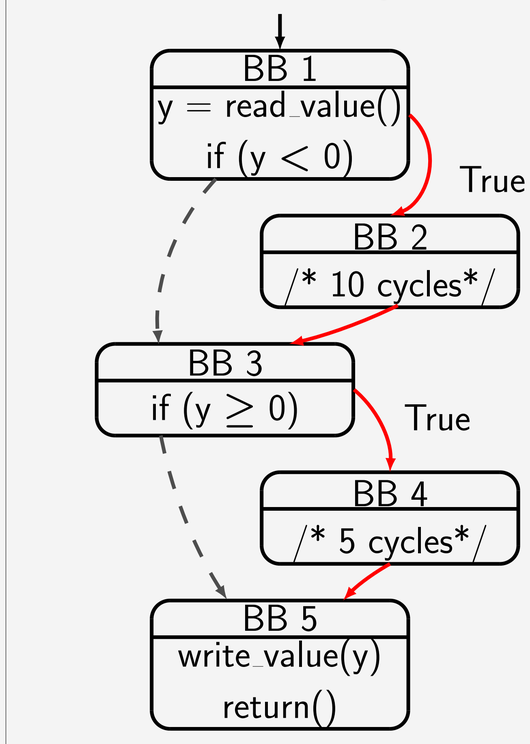
WCET Analysis of Real-Time Systems with Shared TDMA Buses

Existing Techniques to Estimate the Worst-Case Execution Time (WCET)

Analysed Program

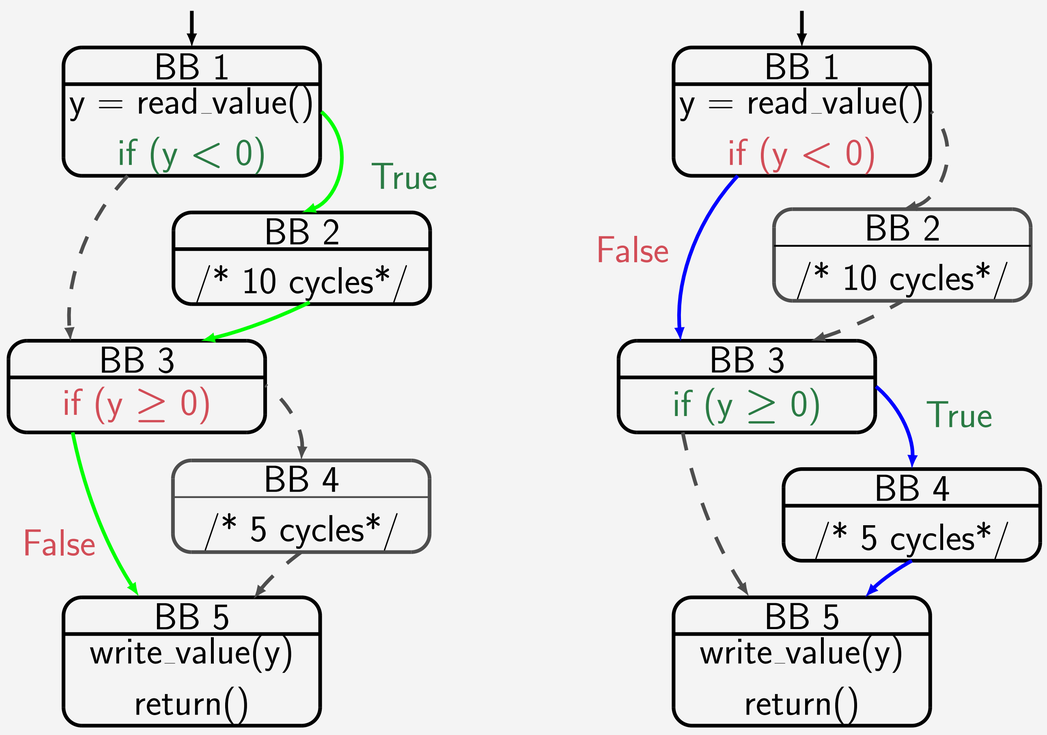


Worst-case Path Without Feasible Path Analysis



WCET=$\underline{BB1 + BB2 + BB3 + BB4 + BB5}$

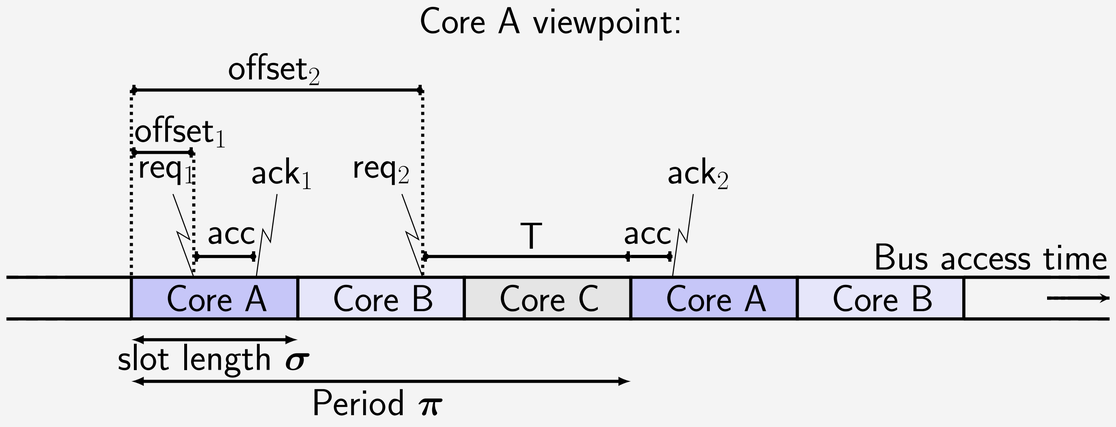
Worst-Case Path With Semantic Analysis



WCET=max{$\underline{(BB1 + BB2 + BB3 + BB5)}$,$\underline{(BB1 + BB3 + BB4 + BB5)}$}

Contribution: Combine semantic analysis and shared TDMA bus analysis to enhance the WCET estimation

1. Time Division Multiple Access (TDMA)



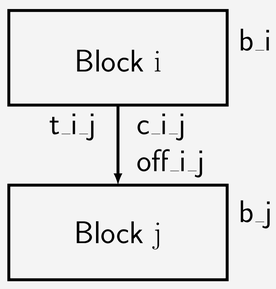
- TDMA: a time triggered arbitration policy for a shared bus
- *acc* access time of granted requests
- Waiting time to grant requests T $\in [0, \boldsymbol{\pi} - (\boldsymbol{\sigma} - \mathrm{acc})[$
- *Offset* = *absolute_time* $\mod$ $\boldsymbol{\pi}$

  Pessimistic approach: *Worst-Case(T)* = $\boldsymbol{\pi}$ - ($\boldsymbol{\sigma}$ - *acc*)

2. Satisfiability Modulo Theory for WCET

↪ **Satisfiability Modulo Theories** (SMT) encoding:

- *b_i "true"* if the *basic block* i is executed
- *t_i_j "true"* if the *transition* block i $\longrightarrow$ block j is taken
- *c_i_j execution time* at the transition block i $\longrightarrow$ block j
- *off_i_j offset* at the transition block i $\longrightarrow$ block j



↪ **Assumptions**:

- ✓ No loops. No recursive function calls
- ✓ Fully Timing Composable Architectures

"How to compute worst-case execution time by optimization modulo theory and a clever encoding of program semantics" Henry et al. LCTES2014

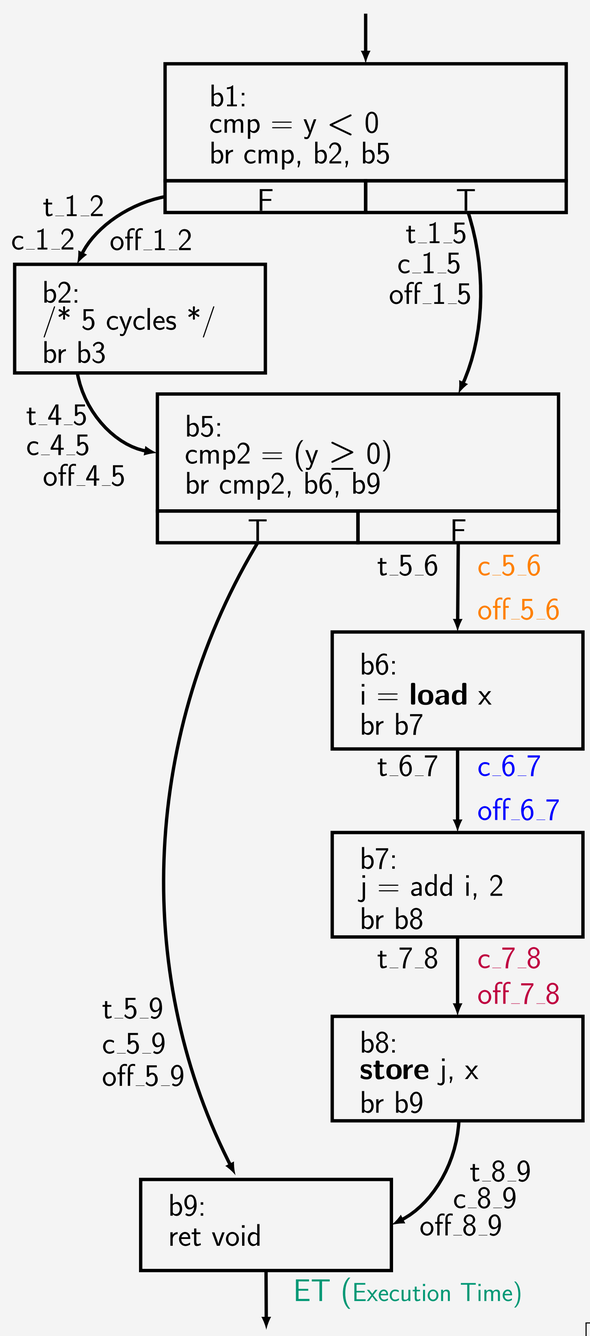
3. Satisfiablity Modulo Theory for TDMA

define-fun **tdma_access** (T_{entry}, offset)
{
  if (0 $\leq$ offset $\leq$ ($\boldsymbol{\sigma}$ - acc))
    return T_{entry} + acc
  else
    return T_{entry} + ($\boldsymbol{\pi}$ - offset) + acc
}

define-fun **tdma_offset** (offset)
{
  if (0 $\leq$ offset $\leq$ ($\boldsymbol{\sigma}$ - acc))
    return offset + acc
  else
    return acc
}

4. Proof of Concept on an Example

- Each *load* and *store* requests an access to the shared bus.
- *load* and *store* are put in separate basic blocks.
- Control Flow Graph obtained with the LLVM framework.



SMT Formula:

(cmp = (y < 0))
$\wedge$*(t_1_5 = b_1 $\wedge$ cmp)*
$\wedge$*(t_1_2 = b_1 $\wedge$* $\overline{\mathrm{cmp}}$*)*
$\wedge$ ...

$\wedge$*(b_5 = t_1_5 $\vee$ t_4_5)*
$\wedge$...

$\wedge$*(c_6_7 =* **tdma_access**(*c_5_6,off_5_6*))
$\wedge$*(off_6_7 =* **tdma_offset**(*off_5_6*))
$\wedge$...

$\wedge$*(c_7_8 = c_6_7 +* $\mathrm{T_{b7}}$*)*
$\wedge$*(off_7_8 = (off_6_7 +* $\mathrm{T_{b7}}$*)* $\mod$ $\boldsymbol{\pi}$*)*
$\wedge$...

$\wedge$*(ET = (ite t_8_9 c_8_9 c_5_9))*

Using an SMT-solver,
Find the smallest X:
ET> X is $\underline{\text{unsatisfiable}}$.

★ Results:

- $\boldsymbol{\pi} = 6$, $\boldsymbol{\sigma} = 2$, 1 instruction = 1 cycle
- Pessimistic Approach **19 cycles**
  Offset-based Approach **15 cycles**

$\underline{\text{Hamza Rihani}}$, Matthieu Moy, Claire Maiza
{first.last}@imag.fr

This work has been funded by grant CAPACITE (PIA-FSN2 n°P3425-146798) from the French Ministère de l'économie, des finances et de l'industrie.