

طراحی افزونه و مد Debugger برای پردازنده RISC-V



دانشجو: محمد تقی زاده گیوری
استاد راهنما: زین العابدین نوایی شیرازی
دانشکده مهندسی برق و کامپیوتر، دانشگاه تهران

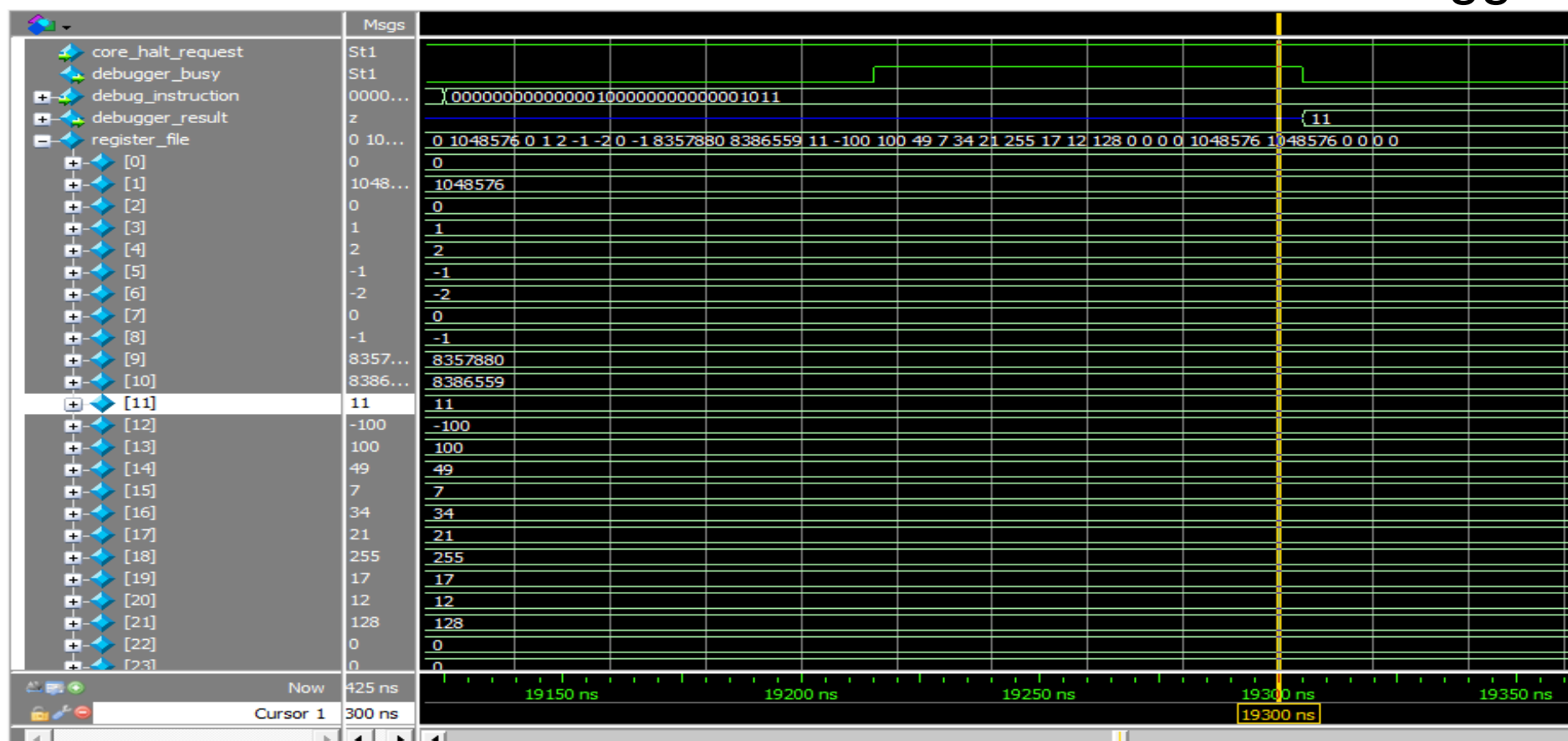
مقدمه / خلاصه

هدف از این تحقیق، ایجاد قابلیت مشاهده روند اجرا برنامه در پردازنده است تا با ایجاد این امکان، قابلیت مشاهده مقادیر ذخیره شده در پردازنده در هر قسمت از برنامه ای که در پردازنده اجرا می شود، وجود داشته باشد تا در این صورت، قابلیت تشخیص منشاء خطا در برنامه تسهیل گردد.

در نتیجه Debugger باید بتواند:

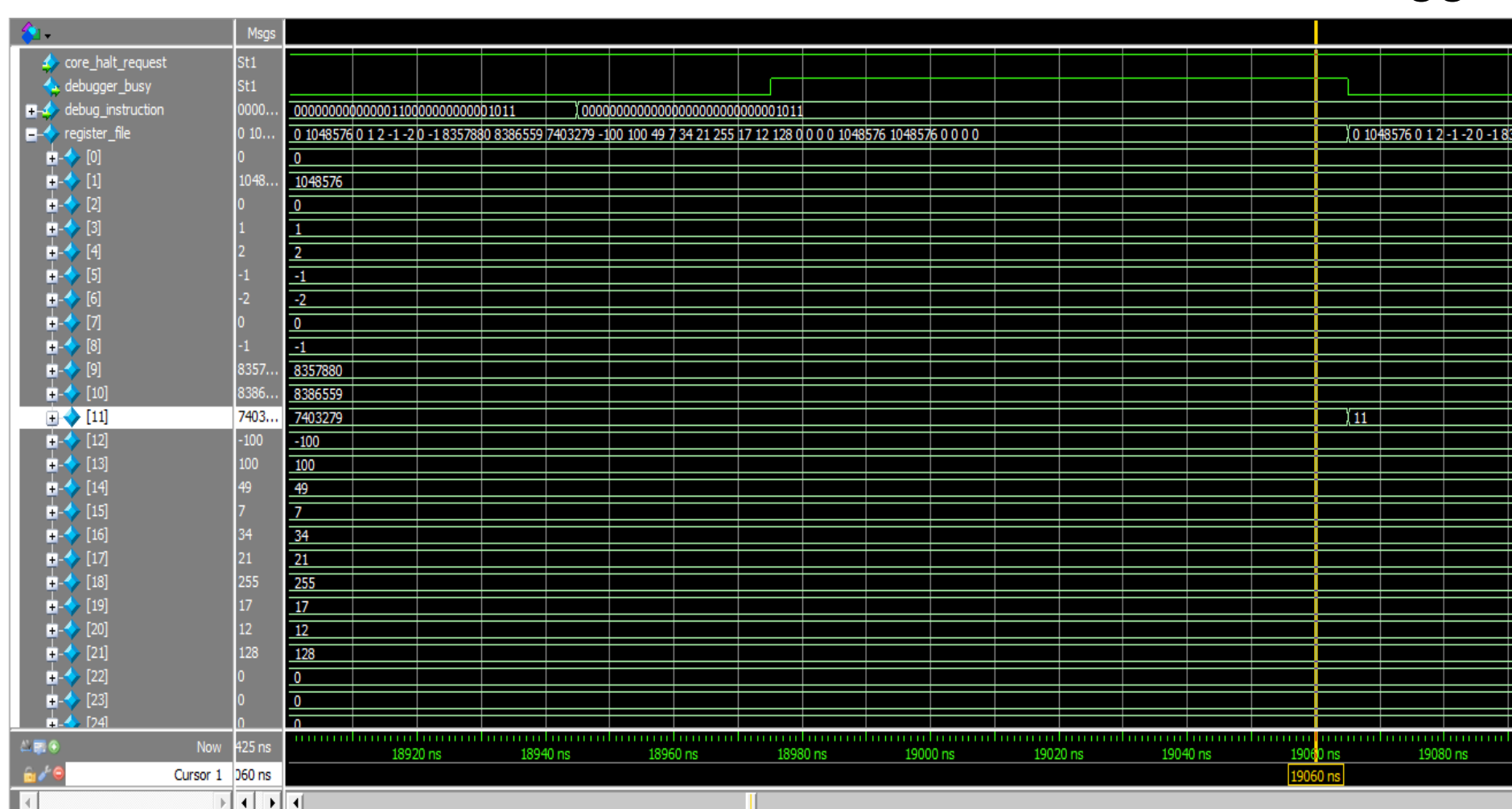
- اجرا برنامه را در هر قسمتی از برنامه، متوقف (pause) کند.
- در حین اینکه برنامه متوقف شده است، مقادیر ذخیره شده در حافظه خارجی و ثبات های حافظه داخلی پردازنده را خوانده و به برنامه نویس نمایش دهد.
- در حین اینکه برنامه متوقف شده است، مقدار مشخصی که از سوی برنامه نویس تعیین شده است را در حافظه خارجی یا ثبات های حافظه داخلی پردازنده ذخیره کند.
- برنامه را، از جایی که متوقف (pause) شده بود، ادامه داده و به اصطلاح resume کند.

خواندن از register file با ارسال درخواست خواندن از register file به Debugger:



همان طور که انتظار میرفت، محتوای ثبات 11 از register file، که با توجه به تصویر فوق، برابر 11 است، بر روی باس debugger_result قرار گرفته که نشان از درستی عملکرد Debugger می باشد.

نوشتن به register file با ارسال درخواست نوشتن به register file به Debugger:



همان طور که انتظار میرفت، در ثبات 11 از register file، مقدار 11 نوشته شده که نشان از درستی عملکرد Debugger می باشد.

جمع بندی

در این تحقیق، ابتدا به طراحی Debugger پرداخته و تغییرات اعمال شده به پردازنده RISC-V را توضیح دادیم. در آخر Debugger طراحی شده را در محیط Verilog پیاده سازی کرده و آن را در حضور پردازنده RISC-V تغییر یافته تست کردیم و مطابق با معیارهای ارزیابی خود، صحت کارایی آن را بررسی کرده و نتیجه گرفتیم که Debugger طراحی شده به درستی طراحی شده و کار می کند.

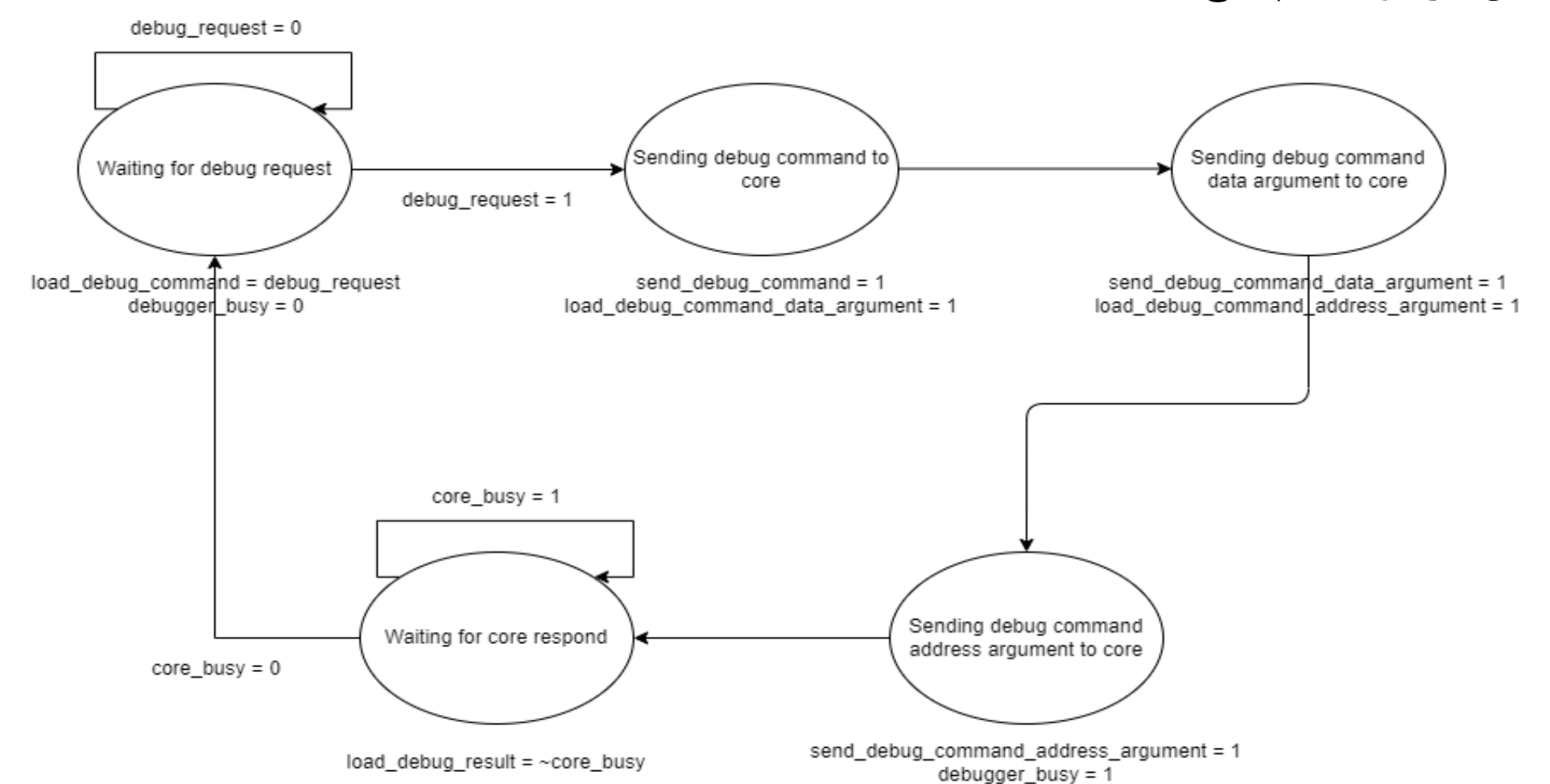
مراجع اصلی

1. Tim Newsome and Megan Wachs, eds., "RISC-V External Debug Support, Version 0.13.2", RISC-V Foundation, 2019.
2. Andrew Waterman and Krste Asanovic, eds., "The RISC-V Instruction Set Manual, Volume I: Unprivileged ISA, Version 20191213", RISC-V Foundation, 2019.
3. Andrew Waterman and Krste Asanovic, eds., "The RISC-V Instruction Set Manual, Volume II: Privileged Architecture, Version 1.10", RISC-V Foundation, 2017.

روش/ساختار/مدل پیشنهادی

با توجه به اهداف و ویژگی های مورد نظر برای Debugger:

1. ابتدا در افزونه Debugger، مطابق با کنترل کننده زیر، درخواستی که از پردازنده داریم را به پردازنده RISC-V طی مراحل زیر ارسال کرده و صبر می کنیم تا پردازنده به آن درخواست پاسخ دهد:



2. پس از ارسال شدن درخواست به پردازنده، لازم است تغییراتی در پردازنده RISC-V اعمال کنیم تا پردازنده قادر باشد به درخواست، که خواندن یا نوشتن به حافظه داخلی پردازنده (register file) یا حافظه خارجی است، پاسخ دهد:

