UC San Diego
Jacobs School of Engineering

Mohsen Imani <moimani@eng.ucsd.edu>

Fwd: Questions abou t your DATE 2016 paper

1 message

Mohsen Imani <moimani@eng.ucsd.edu> Sun, Jul 24, 2016 at 4:59 PM
To: Mohsen Imani <moimani@eng.ucsd.edu>

On Wed, Jun 1, 2016 at 10:14 AM, Xunzhao Yin <Xunzhao.Yin.16@nd.edu> wrote:

Hi Mohsen,

I was wondering if below the model you used for your simulation?

https://nano.stanford.edu/stanford-rram-model

I also know that Prof. Sayeef's group of UCB is fabricating FE devices, and the concept of the FE devices is very similar. It would be great if we could discuss this topic in terms of the collaboration or communication.

Thanks,
Xunzhao

On Mon, May 30, 2016 at 2:47 AM, Mohsen Imani <moimani@eng.ucsd.edu> wrote:

Hi Xunzhao,

No I didn't draw the layout, but based on the technology pitch we estimated the length of the interconnect (my colleague). Also some papers report that size and for other CAM size you can estimate the length. Anyway, if you are working on small CAM size, I you can neglect the interconnect delay or use any simple model. The interconnect impact is much obvious in large CAMs that we do not usually need them for approximate computing. For device model, you can use the released model in Philip Wong group at Stanford. I mostly use their latest ReRAM Verilog-A model. We have also collaboration with UC Berkeley and use the Ferroelectric devices that they fabricate for our simulation. Our target is now NVMs for ultra-efficient neuromorphic computing.
If you are interested in the same topic, we may be able to collaborate in near future.

Regards,
Mohsen

On Sat, May 28, 2016 at 11:21 AM, Xunzhao Yin <Xunzhao.Yin.16@nd.edu> wrote:

Hi Mohsen,

I noticed that the transmission line (interconnect model) command in hspice requires length of line as a parameter, which is assumed to be the size of the layout of TCAM. I was wondering if you drew the layout of your TCAM design and then measure the width and height of the layout to fit the transmission line length? It would be good that if you are convenient to send me a netlist example..

Also I want to do a TCAM comparison among ReRAM, CMOS and the device I am using (called Ferroelectric FET). I was wondering how could I get the ReRAM model to do the simulation?

Thanks,
Xunzhao

On Fri, May 27, 2016 at 2:16 AM, Mohsen Imani <moimani@eng.ucsd.edu> wrote:

Hi Xunzhao,

Yes, the Hspice supports simple RC interconnect model where interconnect model can impact on both delay and energy consumption of TCAM. For measurement we didn't separate the delay of interconnect and TCAM block. We just measure total search delay and energy consumption of TCAM considering all interconnect model. For Hspice Command line please take a look at the following file (page 105-107):
http://www.ece.rochester.edu/courses/ECE222/hspice/hspice_simanal.pdf

Thanks,

Mohsen

On Wed, May 25, 2016 at 10:29 AM, Xunzhao Yin <Xunzhao.Yin.16@nd.edu> wrote:

Hi Mohsen,

Thanks!

I was wondering did you just add the intrinsic wire delay to the HSPICE simulation delay result to get the total TCAM array delay? I was wondering how did you measure each part of the delay?

Second, have you considered the interconnect energy consumption based on the RC model?

Thanks,
Xunzhao

On Wed, May 25, 2016 at 2:53 AM, Mohsen Imani <moimani@eng.ucsd.edu> wrote:

Hi Xunzhao,

Sure! As I mentioned before, we used simple RC model for interconnect modeling. You can model the delay as: *Intrinsic wire delay = 0.5*RC * L^2*
where R and C are wire resistance and capacitance per *um* and the L is the wire length.
The wire delay of each technology is listed in the following file (slide 30):
http://web.stanford.edu/class/archive/ee/ee371/ee371.1066/lectures/lect_09_2up.pdf

Regards,
Mohsen

On Tue, May 24, 2016 at 9:38 AM, Xunzhao Yin <Xunzhao.Yin.16@nd.edu> wrote:

Hi Mohsen,

I was wondering if you are convenient to give me an access to the interconnect model? I think single TCAM row is not enough for the architecture level benchmarking, but I've never done similar simulation before, so I need to have a try.

Thanks,
Xunzhao

On Tue, May 24, 2016 at 12:35 AM, Mohsen Imani <moimani@eng.ucsd.edu> wrote:

Hi Xunzhao,

The impact of interconnect delay is more obvious on the CAM size rather than a single CAM row. Specially when the CAM row is short (<8-bit) the impact of interconnect is negligible. So, your simulation is correct!

Thanks,
Mohsen

On Mon, May 23, 2016 at 11:22 AM, Xunzhao Yin <Xunzhao.Yin.16@nd.edu> wrote:

Hi Mohsen,

I was wondering if the delay and energy consumption of interconnect in small TCAM size are as dominant as in large sizes? Since I didn't include the interconnect model in a single row TCAM with i.e. 8 bits, its result may be a bit different.

Thanks,
Xunzhao

On Fri, May 20, 2016 at 12:16 AM, Mohsen Imani <moimani@eng.ucsd.edu> wrote:

You are right, Just ML voltage reduces due to ML leakage and it has impact on search energy (but not very significant).

On Thu, May 19, 2016 at 8:41 PM, Xunzhao Yin <Xunzhao.Yin.16@nd.edu> wrote:

Hi Mohsen,

For the second note, I am a bit confused. Did you use various VDD? I was thinking that the lower ML voltage is caused by the leakage current, but meanwhile other parts are still connected to VDD, thus the search energy should not be much different..

Xunzhao

On Thu, May 19, 2016 at 11:16 PM, Mohsen Imani <moimani@eng.ucsd.edu> wrote:

Hi Xunzhao,

Yes, that's right!
Yes, also in first cycle the ML voltage is high, meaning that the search energy is higher than other cycles that the ML voltage is lower. Lower ML needs lower search energy regardless of precharging...

Regards,
Mohsen

On Thu, May 19, 2016 at 2:02 PM, Xunzhao Yin <Xunzhao.Yin.16@nd.edu> wrote:

Hi Mohsen,

Do you mean that you measured the energy consumption for i.e. 4 cycles and then get the average energy per cycle by dividing by 4?

For the second note, do you mean that because in the first search TCAM needs to precharge, thus the energy consumption include the precharge energy, besides buffer, interconnect and SA energy consumption and is higher than latter cycles?

On Thu, May 19, 2016 at 4:41 PM, Mohsen Imani <moimani@eng.ucsd.edu> wrote:

Hi Xunzhao,

The energy decrease because we avoid precharging for several cycles! The cycles in the x-axis shows the number of cycles that we can neglect precharging the whole CAM structure. Note that we precharge the hit row selectively (if there is any) but its just on selected line with very low impact on total CAM search energy. As you pointed, in figure 3 we neglect this hit on energy of CAM.
The second note that you should know is that, all cycles do not have the same energy consumption. The first search usually consumes more energy than second and third cycles, because the ML voltage starts discharging by each search cycle. However, before 4-cycles search, ML voltage is high enough to guarantee the exact search operation.

Hope it clarifies
Regards,
Mohsen

On Thu, May 19, 2016 at 12:03 PM, Xunzhao Yin <Xunzhao.Yin.16@nd.edu> wrote:

Hi Mohsen,

5. I don't understand why the energy decreases vs number of cycle in figure 3? Did you assume that during the multiple cycles the TCAM doesn't have any hit, and the ML voltage is gradually leaking current? When you calculated the energy for a certain number of cycles, you just measured the initial energy at the first cycle and the left energy at ML at the last cycle, and the difference between this two energy is the precharging energy for this certain refresh cycle number. Is that the case?

Thanks,
Xunzhao

On Tue, May 17, 2016 at 4:18 PM, Mohsen Imani <moimani@eng.ucsd.edu> wrote:

Hi Xunzhao,

3. the circuit simulation proves that CAM under voltage overscaling or with lower ML voltage works inaccurately with a few number of mismatches. Because CAM in lower

ML voltage works slower, meaning that it can not distinguish 1-bit mismatch and exact matching. Lower supply voltage will further increase this number of mismatches. I think the DATE'15 paper (which I sent you before) is the best paper to understand why TCAM hits approximately under voltage overscaling.

4. Yes! I forgot to mention here that after each search cycle, we selectively precharge a row(s) which data matched there. But not complete precharging on all MLs. This selective precharging helps us to avoid consuming large energy for whole CAM precharging. Because of this reason we do not consider the performance improvement which can happen by multiple-cycle search. If we can neglect this selective precharging, we can have very fast search operation by continuing the search operation for several cycles. I explained this fact in both conference and journal papers.

5. We used simple circuitry (two series inverter) as an input buffer. The ratio of their size has been set to 4x, meaning that the second invertor has 4x larger size than first one. In large CAMs we need to increase the size of both inverters to guarantee fast search operation. Otherwise with small buffer, distributing the input signals among all rows will take long time (slow search operation).

The evaluation energy cons from input buffer and interconnects and sense amplifier.

To find average energy vs cycles, we just simulate the search operation on a MASC CAM for multiple cycles. Meaning that in first cycle, the ML is charged to VDD. After that search the ML voltage reduces a bit. With that ML value, we did the second search operation, etc. This can give us the correct energy that CAM consumes.

Regards,

Mohsen

On Tue, May 17, 2016 at 7:07 AM, Xunzhao Yin <Xunzhao.Yin.16@nd.edu> wrote:

> Hi Mohsen,
>
> Thanks for your reply, that helps me a lot! I just have a bit more confusions:
>
> 3. I was wondering how did you determine that a CAM row have 2 bits mismatch? Since the 8-bit CAM is using one matchline, you cannot tell from the voltage of matchline that the searching value has 2 bits mismatch with the stored value.
>
> 4. Do you mean that each cycle will do precharge independently if in this cycle the search is a hit? For example, if you are searching for "00000000" and this value is matching with "00000000" stored in CAM in the first cycle, then the matchline will be discharging and drop to zero. Then this CAM cannot work any more until the matchline is precharged again after the refresh cycles.
>
> 5. May I know the circuit of the input buffer you used in your design? And is the buffer the row driver in figure 1? Also I was wondering if the evaluation energy is the energy consumed in the input buffer in every evaluation cycle? Since the CAM circuit itself only consumes energy when precharging, the evaluation energy can only come from the input buffer. Is that correct? In addition, may I know how did you calculate the average energy consumption vs number of cycles?
>
> Thanks for the journal paper, I think I should read it carefully.:)
>
> Xunzhao
>
> On Mon, May 16, 2016 at 6:55 PM, Mohsen Imani <moimani@eng.ucsd.edu> wrote:
>
>> Hi Xunzhao,
>>
>> Sorry! I totally miss your email. Here is my responses to your Qs:

3- What you understood is almost correct. But the idea of HD works for a CAM with a row containing multiple bits. We got the figure 3&4 for a CAM row containing 8-bit (a CAM which has 8-cell in each row not 8 separated cells). 1-HD means that if you do not precharge more than 4-cycle you have possibility of wrong matching (HD). For example, if you do not precharging ML CAM for 6 cycle, you have 1-bit wrong matching on the 6th and 7th cycles. In 8th cycles you may have 2 bits mismatch in total 8-bit. Note that these wrong bit matches can happen in any eight bit of a row (not important where).

4- Each cycle is an independent search. In first search I am checking for input A1+B1=C1 precisely. Similarly, in the next cycles (before cycle 4th) I am searching for **other inputs** precisely. But after 4th cycles there may be a mismatch in your search. For example, if you are searching for “00000000”, this value will not match with “00100000” (store value in CAM) in first cycles. But after 6th cycles, our design will match it approximately, since they have just a bit hamming distance.

5- TCAM energy have two terms: precharging and evaluation. Although we use long time precharging, we need to pay the cost of evaluation for each search cycle. This energy is not negligible in large CAM size because of the interconnect and input buffer energy (which should distribute the input data among all CAM row simultaneously). In addition, when the voltage of ML drops, the TCAM requires less energy for search operation. So, dividing energy by the number of cycles does not make sense. When the CAM size increases, we need to have larger input buffer to distribute the input data among all CAM rows at the same time. The delay of this buffer is a dominant search delay/energy term in the large CAMs. You should consider this fact in your design.

6- No, I considered the delay of both MASC stages to calculate the clock frequency. We use HSPICE to calculate both CAM stages delay for each search operation. As we explained, MASC TCAM is going to be searched in parallel with a single FPU stage delay. So, in architecture level, the clock frequency of FPU stage is set based on the delay of MASC in each size. For example, if I am working with 32-row CAM, I set the clock frequency of FPUs in e.g. 1.5ns, while using 64-row increases the FPU clock cycles to e.g. 2ns.

I attached our accepted journal paper which is the extended version of DATE paper.

Regards,

Mohsen

On Mon, May 16, 2016 at 11:06 AM, Xunzhao Yin <Xunzhao.Yin.16@nd.edu> wrote:

> Hi Mohsen,
>
> Hope I am not bothering you too much. I was wondering whether you available to answer my questions above? I have done my simulations based on the methods I described above, and just want to make sure that I didn't do anything unreasonable.
>
> Thanks,
> Xunzhao
>
> On Thu, May 12, 2016 at 5:40 PM, Xunzhao Yin <Xunzhao.Yin.16@nd.edu> wrote:

6. Also I was wondering how did you determine the search clock period? It seems that you set the search clock period as the maximum TCAM delay at the last period. It sounds to be controversial since you have already set the search clock period before you find the maximum TCAM delay at the last search cycle. Did you just simulate the TCAM for several cycles with a certain clock period and measure the TCAM delay at the last cycle and then set it as the search clock period?

Thanks,
Xunzhao

On Thu, May 12, 2016 at 3:45 PM, Xunzhao Yin <Xunzhao.Yin.16@nd.edu> wrote:

> Hi Mohsen,
>
> Thanks for you response.
>
> 3. Let me figure out the meaning of 1-bit HD first. Assume we have a 8 bit TCAM row (not 8-bit row, but 8 TCAM blocks each has one bit TCAM), is 1-bit HD means that 7 bits are matching while only one is mismatch? Here is my understanding: in a case of 8-bit MASC design, a match results in discharging ML, and a mismatch keeps the matchline high. If exact matching is required, then after 4 search cycles the matchline voltage has leaked and dropped down to a level that may be sensed as '0' (meaning a match), thus a precharge needs to be applied. If 1 HD matching is allowed, then after 6 search cycles one of the 8 matchline voltages drops down to be sensed as '0', and others are still sensed as correct matching. Am I right?
>
> 4. Let me put an example: a TCAM block is refresh every 4 cycles, but at the third search cycle a hit occurs. What would the controller do? Will the controller change the precharge cycle so that the TCAM block is refreshed at the end of the third cycle, or will the TCAM still refresh after 4 cycles? I know this functionality is not correct, I was just wondering how to force the TCAM refresh at the third cycle?
>
> 5. Since I am doing circuit level simulations, I am interested in the average energy consumption in figure 3. Did you just calculate the precharging energy consumption and divide it by the number of different precharging cycles? Also I am a bit confused for the latencies in figure 5. The figure shows that the delay increases as the number of TCAM rows grows. Does this mean that all the TCAM rows are precharged by the same pull-up network? I was thinking that the precharging of TCAM rows are done in parallel by different precharging devices, but this figure indicates that all rows are precharged by one device, that's why the delay increases because the matchline load has grown as row number increases.
>
> Thanks,
> Xunzhao
>
> On Wed, May 11, 2016 at 6:51 PM, Mohsen Imani <moimani@eng.ucsd.edu> wrote:
>
>> I forgot to mention that, recently the extended version of MASC paper has been accepted in IEEE Transaction on Emerging Topics in Computing. It may help you more to understand the MASC functionality. I will share the paper with you shortly after speaking with my advisior.
>> Here is our other DATE paper which is closely related to MASC paper: http://moimani.weebly.com/uploads/2/3/8/6/23860882/recam-date-mohsen.pdf
>>
>> Mohsen
>>
>> On Wed, May 11, 2016 at 3:37 PM, Mohsen Imani

<moimani@eng.ucsd.edu> wrote:

> Hi Xunhao,
>
> Sorry for my late respond. I was busy with my proposal defense.
>
> 2. We use simple RC model for interconnects. The values are extracted from 45nm TSMC technology. If you have TSMC, I recommend you can use the available interconnect model/library that TSMC has. It's more close to real implementation.
>
> 3. Actually understanding the figure 3 & 4 is a little bit hard. I am trying to explain it easier in the following:
>
> Figure 4 shows the output signal of sense amplifier, when we have matching, 1-bit hamming distance (HD), 2-HD, etc. Having more number of mismatches meaning faster discharging current. Now consider another case that we didn't show it in figure. Think we have a TCAM row where input data is exactly match with the input operand, what will happen in the case that we apply voltage overscaling on the TCAM? Will this ML still stay charge? NO! the ML voltage will drop in lower voltage, correct? If you decrease the supply voltage more and more, the ML will discharge faster. When this discharging (due to voltage overscaling) is exactly corresponds to a case that we have 1-bit hamming distance? Similarly, lower supply voltage corresponds to 2-bit HD and so on. The level of voltage overscaling depends on CAM bitline size and type of TCAM cell that you are using.
>
> In a case of MASC design, the hamming distances can be obtained similarly but using a bit different way. In MASC there is no voltage overscaling. However, doing the search operation for multiple stage reduces the ML voltage. Our circuit level simulation shows that the dropped ML voltage is corresponding to have 1-bit, 2-bit ,…. hamming distance.
>
> 4. The overhead of long time controller is consider on HSPICE. For architectural simulation, you don't need to model long time precharging or MASC configuration on multi2sim. The Multi2sim simulator just understands MASC as a TCAM block which can do the search operation exactly or approximately. So, we only model a CAM block beside each FPU (ADD, MUL, MAD, SQRT), fill it with high frequency patterns in profiling and then relax the computation in some of CAM bits in approximate mode. Meaning that we allow the input data "0101" to match with CAM row which stores "0100" when we are in 1-HD mode.
>
> Mohsen
>
> On Wed, May 11, 2016 at 7:15 AM, Xunzhao Yin <Xunzhao.Yin.16@nd.edu> wrote:
>
> > 4. Also I was wondering how did you apply refresh schemes in the Multi2sim simulator? Since the refresh period is a fixed number of cycles while we don't know which cycle the hit will occur, you need to do pre-charging whenever the refresh or hit occurs. This seems to increase the overhead of precharging controller.

On Wed, May 11, 2016 at 10:09 AM, Xunzhao Yin <Xunzhao.Yin.16@nd.edu> wrote:

Hi Mohsen,

2. For the circuit level simulation, could you please tell me the simple model of interconnect you used? Did you just assume a fixed resistance and capacitance at the source/drain of transistors?

3. I am still confused about the simulation you have done for figure 4. In figure 4 the ML voltage finally drops to 0, indicating a mismatch, which is assumed to be an error result, am I right? Also, I don't understand the 1-bit, 2-bit,.... HD. How did you determine that the 8-bit TCAM rows have 1-bit, 2-bit HD? Did you just count the number of rows that have output wrong results?

Thanks,
Xunzhao

On Tue, May 10, 2016 at 11:01 PM, Mohsen Imani <moimani@eng.ucsd.edu> wrote:

Hi Xunzhao,

1. Figure 3, 4, 5 are obtained using HSPICE results. While, figure 6 and 7 are the GPU energy which is the result of architectural simulator + Design compiler + HSPICE.

2. For circuit level simulation, I used simple model to estimate the energy and delay of interconnect in HSPICE. Didn't use complicated interconnect models.

3. I just simulate a MASC row where the input data is missing there for several cycles. From circuit simulation, I figure out the ML discharging current (blue line in figure 3). Then, I have done another set of circuit level simulation to understand the functionality of the MASC under low ML voltage. Generally, CAMs under voltage overscaling can hit with the input data in a few number of hamming distance bits. (for more information please take a look at: http://mesl.ucsd.edu/site/pubs/DATE15_Abbas.pdf). To plot We find the level of ML voltage that is corresponding to 1-bit, 2-bit, 3-bit, … hamming distances, and then fit them with the points of figure 3.

Mohsen

On Tue, May 10, 2016 at 6:03 PM, Xunzhao Yin <Xunzhao.Yin.16@nd.edu> wrote:

Hi Mohsen,

Thanks for your detailed response. I am new in the architecture benchmarking, thus don't understand the evaluation thoroughly.

1. I was wondering which figures are the results obtained

from HSPICE simulations in your paper? Fig. 3, 4, 5?

2. I was wondering did you take the metal capacitance of matchline, searchline, etc. into consideration for the energy and delay of TCAM, or did you measure the energy and delay of TCAM just at circuit level in HSPICE?

3. Speaking of the refresh schemes, I was wondering how did you do the refresh period estimation in simulation (FIg. 3, 4)? Did you assume no match (or no discharging) occurs during the simulation and just plotted the matchline voltage vs time (Fig. 4)?

Thanks,
Xunzhao

On Tue, May 10, 2016 at 7:53 PM, Mohsen Imani <moimani@eng.ucsd.edu> wrote:

> Hi Xunzhao,
>
> I appreciate your interest in our DATE paper.
>
> 1. Yes, we used HSPICE simulator to design a TCAM cell in circuit level. For memristor library you can find several online Verilog-A code in internet. The most recent models are for Philip Wong group in Stanford. After designing the cell, in HSPICE, you need to design a single TCAM row and then extend it to complete TCAM block. In MASC the configuration 2:16 means that we 16 small size TCAM block where each uses 2-bit encoding scheme that we proposed in paper (TCAM bitline needs to contain 32-bit for simple addition in floating point). In architecture, we modify the GPU code to model the MASC TCAM beside each floating point unit in architecture level. We didn't feed any delay or energy to architecture simulator, instead, for each floating point unit the simulator reports the hit-rate of the TCAM blocks and number of usage of each floating point units in exact or approximate matching. To have more realistic result than simulator power modeling, we design 6-stage balanced FPUs using FloPoCo and implement them using 45-nm TSMC technique in Design compiler. Then, based on the MASC search delay, we power optimized the FPUs using IC compiler. Finally based on extracted statistics from simulator, ASIC FPUs energy report and HSPICE MASC energy report, we manually estimate the energy consumption that GPU can achieve.
>
> 2. Yes, we used HSPICE simulation to calculate the energy and delay of MASC TCAM in different sizes. Therefore, all advantages of ReRAM, such as low energy, high density, etc. are already considered in our simulations.
>
> Please feel free to contact me in case of any question, I would be happy to answer you,

Regards,

Mohsen

On Tue, May 10, 2016 at 8:45 AM, Xunzhao Yin <Xunzhao.Yin.16@nd.edu> wrote:

Hi Mohsen,

Hope I am not bothering you too much. I am Xunzhao Yin, from University of Notre Dame, Dr. Sharon Hu's group. I have read your paper about the MASC strategy for ReRAM TCAM design.I am very interested in the circuit level simulation for the TCAM, since I am also working on the evaluation of some TCAM cells. Now I have a couple of questions about the TCAM evaluation, and I was wondering if you are available to answer them:

1. I was wondering what size of TCAM circuits did you simulate using HSPICE? Did you simulate a single TCAM cell or different MASC TCAM array sizes based on different MASC configurations (i.e. 2:16, 4:8, 8:4)? I assume you got the delay and energy of HSPICE simulation and fed it to the architecture level simulator (Multi2sim). Am I right?

2. The ReRAM TCAM has much smaller area overhead than conventional CMOS TCAM due to the advantages of ReRAM. I was wondering if you have considered the impact of area reduction on the delay and energy at architecture level evaluation? Since the wordline, matchline load (length, capacitance, resistance) have changed compared with conventional CMOS TCAM.

It would be of great help if you could help me figure out the questions, and I am looking forward to you reply!

Thanks,
Xunzhao Yin

--
Mohsen Imani
PhD Student,
Department of Computer Science and Engineering,
University of California San Diego, US.

--
Mohsen Imani
PhD Student,
Department of Computer Science and Engineering,
University of California San Diego, US.

--
Mohsen Imani
PhD Student,
Department of Computer Science and Engineering,
University of California San Diego, US.

--
Mohsen Imani
PhD Student,
Department of Computer Science and Engineering,
University of California San Diego, US.

--
Mohsen Imani
PhD Student,
Department of Computer Science and Engineering,
University of California San Diego, US.

--
Mohsen Imani
PhD Student,
Department of Computer Science and Engineering,
University of California San Diego, US.

--
Mohsen Imani
PhD Student,
Department of Computer Science and Engineering,
University of California San Diego, US.

--
Mohsen Imani
PhD Student,
Department of Computer Science and Engineering,
University of California San Diego, US.

--
Mohsen Imani
PhD Student,
Department of Computer Science and Engineering,
University of California San Diego, US.

--

> > > > > > > > Mohsen Imani
> > > > > > > > PhD Candidate,
> > > > > > > > Department of Computer Science and Engineering,
> > > > > > > > University of California San Diego, US.
> > > > > > > > Webpage: http://moimani.weebly.com/
> > > > > > >
> > > > > > >
> > > > > >
> > > > > > --
> > > > > > Mohsen Imani
> > > > > > PhD Candidate,
> > > > > > Department of Computer Science and Engineering,
> > > > > > University of California San Diego, US.
> > > > > > Webpage: http://moimani.weebly.com/
> > > > >
> > > > >
> > > >
> > > > --
> > > > Mohsen Imani
> > > > PhD Candidate,
> > > > Department of Computer Science and Engineering,
> > > > University of California San Diego, US.
> > > > Webpage: http://moimani.weebly.com/
> > >
> > >
> >
> > --
> > Mohsen Imani
> > PhD Candidate,
> > Department of Computer Science and Engineering,
> > University of California San Diego, US.
> > Webpage: http://moimani.weebly.com/

--
Mohsen Imani
PhD Candidate,
Department of Computer Science and Engineering,
University of California San Diego, US.
Webpage: http://moimani.weebly.com/

--
Mohsen Imani
PhD Candidate,
Department of Computer Science and Engineering,
University of California San Diego, US.
Webpage: http://moimani.weebly.com/

--
Mohsen Imani
PhD Candidate,
Department of Computer Science and Engineering,
University of California San Diego, US.
Webpage: http://moimani.weebly.com/

--
Mohsen Imani
PhD Candidate,
Department of Computer Science and Engineering,
University of California San Diego, US.
Webpage: http://moimani.weebly.com/