IMPLEMENTATION OF EFFICIENT STREAMING APPLICATION USING CLOCK GATING

[1]Lankapalli Manasa, [2]Dr.R.S Rao

[1] M.Tech student (P.G scholar), [2] Professor

[1](Department of Electronics and Communication Engineering)

[1](Gudlavalleru Engineering College, Gudlavalleru India-521356)

Abstract: This paper introduces the decrease in vibrant energy generated by asynchronous dataflow models using clock gating methods for streaming apps. Streaming apps are a very wide range of computing algorithms in fields like signal processing, electronic press coding, cryptography, audio analytics, network routing, message handling, etc. In semiconductor microelectronics, clock gating is a power-saving function that allows to switch off circuits. This article presents circuit gating methods that can accomplish energy benefits by selectively turning off components of the systems when they are momentarily idle, taking into account the vibrant streaming conduct of computers Application-independent methods can be introduced to any implementation and incorporated into the synthesis phase of a high-level dataflow design flow. Experimental findings indicate that no loss of information throughput is accomplished.

Keywords: Clock-gating, dataflow, high- level synthesis.

I. INTRODUCTION

The current significant restriction of silicon computing systems is power dissipation. It also means less stringent heating requirements, enhanced lifespan, greater independence for battery-operated appliances and clearly reduced energy expenses. Power also commonly impacts the decision of the computing platform straight from the beginning for all these purposes. Field-programmable door panels (FPGAs), for instance, indicate greater energy dissipation per memory device relative to similar application-specific embedded loop (ASIC), but often contrast positively with standard computers used for the same operational functions. Power dissipation can be divided into two parts for any semiconductor unit:

1) Static Component.

2) Dynamic component.

The outcome of the leakage flow of the transistors, also impacted by the ambient temperature, is static energy dissipation, also referred to as quiescent or standby power consumption. Dynamic power dissipation, on the other hand, is triggered by switching transistors and drops of fees moving along cables. Power dissipation improves linearly with speed, mainly owing to parasite capacitance impact. In the last 20 years, ASIC developers have used computer gating (CG) methods to counteract this impact. Various policies for optimizing ASIC and FPGA power consumption. In semiconductor mainly in the microelectronics, clock gating is a power-saving function that allows to switch off circuits. To decrease vibrant power consumption, many electronic systems use clock gating to switch off trains, switches, fences and processor components. It defines the effect on a specified architecture of a selected technology, but does not explain how to decrease authority at the stage of construction abstraction. Consequently, the addition of energy amplifiers at the cognitive description design level is an extra job to be carried out with regard in order to prevent the introduction of unwanted implementation features and may decrease software portability (i.e., device is modified during the implementation process). GALS works can be divided into three classifications:

1) Partitioning ;

2) Equipment for communication ; and

3) Dedicated architectures.

Modeling, discovery and development of dataflow models for GALS-based models was earlier researched by several writers. Dynamic dataflow models such as those expressible using the official RVC-CAL language have exciting properties that can be used to reduce power consumption without influencing the application's cognitive features through building. In RVC-CAL, each actor can perform handling duties simultaneously, trials can be disabled by preventing entry passes, and any communication between performers can happen only through exchange maintaining lossless queues. As a result, an actor can be halted for a certain duration if its handling activities are inactive or its output queues (buffers) are complete without affecting the design's general performance and semantic behavior.

II.CLOCK GATING TECHNIQUE

The scheme that processes parts on a constant stream is called the streaming protocol. Example Digital audio coding, cryptography, image analysis, network routing, handling of packets, etc. By adopting video analytics (MPEG-De-blocking filter) as an instance to stream request here. The clock gating of streaming implementation is shown in Fig.1.

Problem in streaming applications

1) Due to continuous-direct clock entry, streaming implementation vibrant power consumption is higher.
2) More heat dissipation due to consumption of electricity.

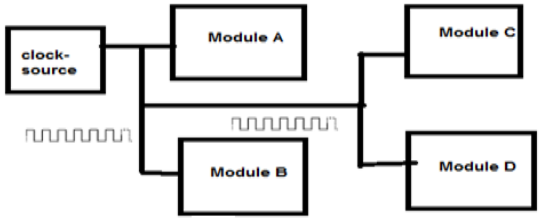

Fig.1: Streaming application with direct clock gating

Remedy

Power gains can be achieved by selectively turning off circuit components when they are momentarily idle.
Steps:1) Initially monitor all components in the scheme (A, B, C, D and E).
2) Identify a module that is not in use momentarily (F, AF feedback inputs).
3) Turn the unit off by disabling the entry of the clock.
4) Clock gating can be accomplished.

III. PROPOSED SYSTEM

In this case, the input clock signal should not be given to the modules inside the application system directly. By bringing the clock through a gate or buffer, it is referred to as clock gating as shown in Fig.2.

1) It is provided through the allow signal (EN) regulated doors.
2) If EN=1 enables a specific module to receive a clock signal.
3) Else (EN=0) stops the clock entry for the module in question.
4) It is necessary to drive this allow (EN) from the clock enabler loop.

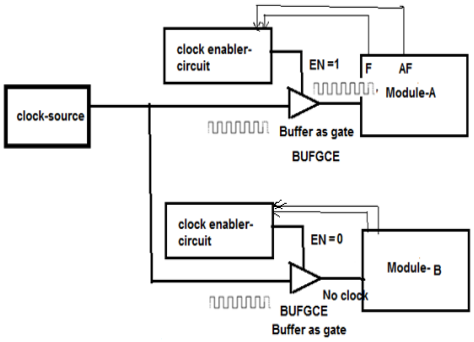

Fig.2:Streaming application without direct clock gating

Implementation

To prevent device feedback to the system by selecting EN=0 by discovering that any device is in inactive state (get in start mode). Then turn off the dormant provisional unit.

1) Clock-gating prevents useless module clocking.
2) Every time a module requires a clock, it only provides the clock otherwise.
3) This decreases the vibrant power consumption of the streaming implementation by clock-gating.

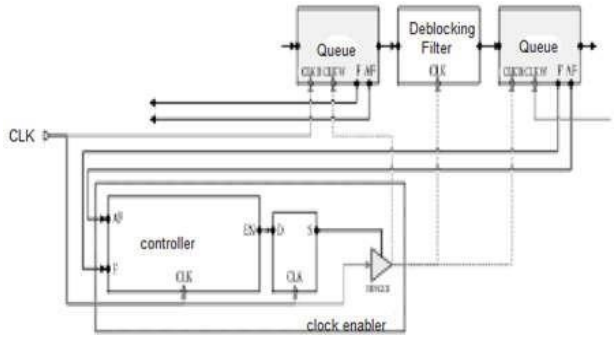

Fig.3: CG methodology applied for actor A.

Here in place of actor-A (module-A) taking any kind of circuit as shown in Fig.3. For example Let us take a De- blocking filter as actor-A. It is a part of MPEG encoder and filters image frames of video signal.

Operation of clock-gating technique

1) From figure, first queue stores the pixel data of the image and De-blocking filter takes it as input filter the data of the image. Then filtered output immediately stored onto the second queue. Later on data will be stored on to the main memory or send for the broadcasting.
2) When second queue is filled with FULL of data or ALMOST-FULL there is no need of filter (ACTOR-A) working, it should be off until second queue has some empty space to store data.
3) To avoid overwriting of data in second queue or to reduce the dynamic power consumption and to stop the clocking of Actor-A, second queue and first queue.
4) To stop clock signal and giving EN=0 to the clock buffer. EN has determined by clock enabler circuit which senses the F, AF signals from second queue.
5) There are so many modules or ACTORS in MPEG encoder; switch off the modules which are temporarily in not-working mode by stopping the clock to them.
6) Using clock gating technique (clock enabler circuit + clock buffer) we can reduce the dynamic power consumption of streaming application which has lot of modules.

The state diagram of clock enabler circuit is shown in Fig.4.

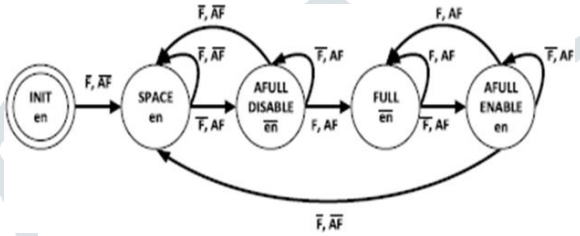

Fig.4: State diagram of clock enabler circuit

1) Clock enabler panel state diagram Clock enabler loop uses F, AF signals as second queue outputs and provides EN as input.
2) If F=0, AF=0 implies that it is not FULL (F, AF) for the second queue and that it has some room to display information. In this case, by enabling gate or buffer (in mean en=1), we must provide the clock to the circuit.
3) If F=1, AF=1 (F, AF) implies second queue is FULL and there is no room for data storage.The enabled clock circuit provides EN=0. (means en=0) performance.
4) The output(s) of the clock enabler system depends on outputs (F, AF).

The internal operation of de-blocking filter is shown in Fig.5.

Fig.5:Internal operation of De-blocking Filter

It requires video frame lateral pixel values to filter H1, H2 (filter unit1, unit2) and image picture vertical pixel values to filter V1,V2(filter unit3,unit4).

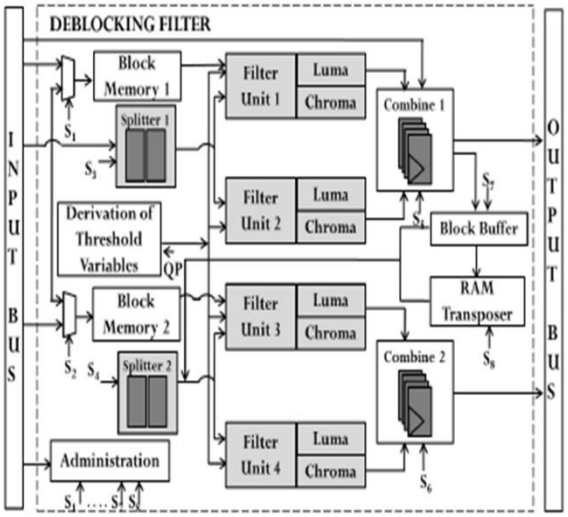

Fig.6:Parallel De-blocking Filter as an Actor unit

In this architecture there are five primary elements as shown in Fig.6. The first component is the two memory blocks that store the received data from an external memory. Each storage is a dual-port SRAM of 32x32 bits. Block Memory 1 is used to place Filter Unit 1's remaining neighboring frames(E1-E8). Block Memory 2 is used to place Filter Unit 3's bottom neighboring layers(F1-F8).

The second component is Filter Units with two horizontal filters and two vertical filters. Two lateral filter modules, HF1 and HF2, are Filter Unit 1 and Filter Unit 2. Two vertical filter modules, VF3 and VF4, are Filter Unit 3 and Filter Unit 4. Furthermore, the de-blocking filter architectures for luminance and chrominance are divided and simultaneously performed. The following element is the derivation of limit factors from the initial quantization producing limit factors and tc and the Parameter(QP) setting. Additionally, a RAM Transposer that transposes the information stream of 8x8 frames from row to column to assist reuse the linear filter information for the vertical filter without holding away to the database.

DOUBLE EDGED TRIGGERD FLIPFLOP

Purpose of DET FF

1) The velocity of information transactions in microprocessor registers with multi-bit FFs can be increased.
2) SET FF transfers 3 parts per 3 pulses of the clock.
3) But for the same 3 clock pulses, DET FF transfers 6 points.
4) Use less DET-FFs instead of more SET-FFs if we keep the same data rate.

The speed of data operations with multi-bit FFs in microprocessor registers can be improved. SET FF transports three components per three clock pulses. But DET FF exchanges 6 marks for the same 3-clock signals. If we maintain the same data rate, use less DET-FFs instead of more SET-FFs.

The key to implementing the above methodology is to use a double-edge triggered flip flop (DET FF) that can work on both edges of the clock instead of using the conventional single-edge triggered flip-flop (SET FF), as shown in Fig.7.DET flip-flop, which has not yet been reported to be used in any server processors in the data center.

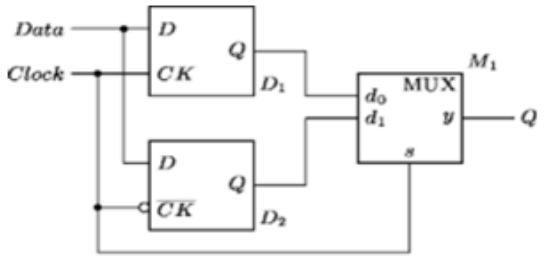

Fig.7: Double-Edge triggered Flip-Flop

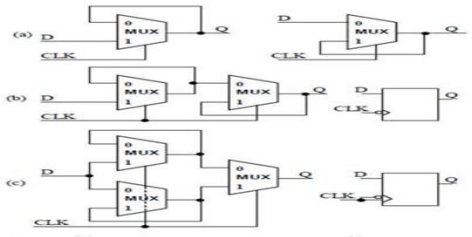

Fig.8:(a) Positive and negative level sensitive-latches ;(b) SET flip-flop (c) DET flip-flop

Because the condition of the flip-flop may change at both falling and rising edges of the clock, it is called ' Double-Edge-Triggered Flip-Flop as shown in Fig 8.

LOGIC STRUCTURE OF A DET FLIP-FLOP

Latch is the fundamental organ in which a flip flop is composed. A clock rates, CLK, are used to guide the latch either to the state of space or to the state of entry. The state equations for beneficial and bad level-sensitive lock can be displayed as:

$$Q = D.CLK + Q.CLK \quad (1)$$

$$Q^1 = D.CLK + Q.CLK \quad (2)$$

Equation (1) defines a latch passing the entry information when CLK= 1 and saving it when CLK=0. Inversely, formula Equation (2) defines an additional latch that gets CLK= 0 entry information and shops it at CLK= 1. A MUX can be used to realize the respective logic constructions. The debate above demonstrates that when CLK= 0, the guide lock does not obtain the entry information. Obviously, if both clock rates need to receive the entry information, these two supplementary latches should be linked in tandem rather than in sequence. Since the flip-flop is needed from entry to production to be non-transparent, the production terminal should always be linked to the memory button. The MUX represented by the dotted row is therefore necessary. Because the state of the flip-flop can change its beneficial as well as adverse sides called "Double bladed activated flip-flop. The transient analysis waveforms are shown in Fig.9 and Fig.10.

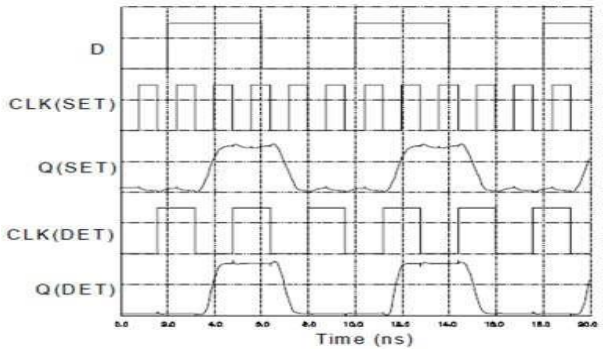

Fig.9: Transient analysis waveforms

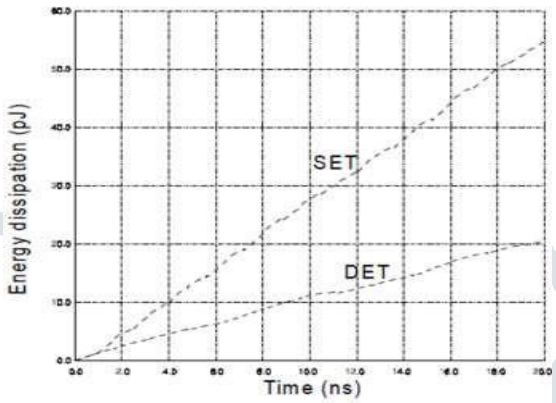

Fig. 10: Energy dissipation diagram

1) DET flip-flop's suggested fresh circuit design has perfect logic function. Compared to the traditional SET flip-flop, it can obtain, test, and retain entry information at the same information speed but at twice the CPU frequency. By using the DET flip-flops, the loop energy is decreased.
2) By evaluating the simulation outcomes of the constructed CMOS DET flip-flop and the traditional CMOS SET flip-flop, the former not only has reduced wait period but also greater peak data rate as its memory button is nearer to the target.
3) By comparing the simulation results of the designed CMOS DET flip-flop with SET, our design was found to have a simpler structure, less delay time and the maximum date rate.
4) By recreating an enhanced DET flip-flop and discovered that the poor power quality of this DET flip-flop is the most preferred.

IV.RESULTS

The simulation waveforms is shown in Fig.11.

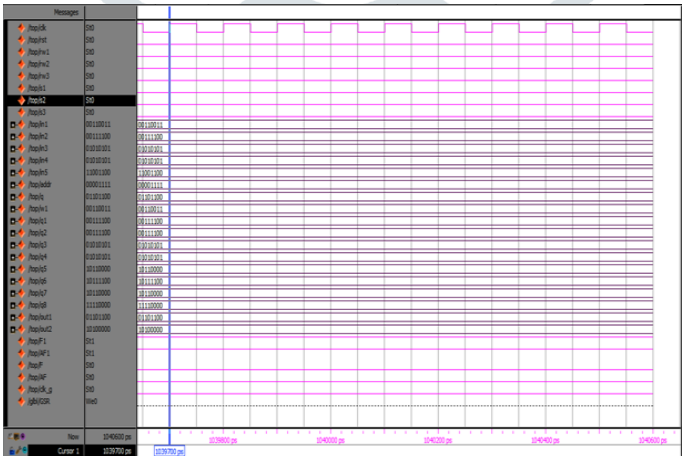

Fig.11: Simulation Waveforms

The RTL schematic is shown in Fig.12.

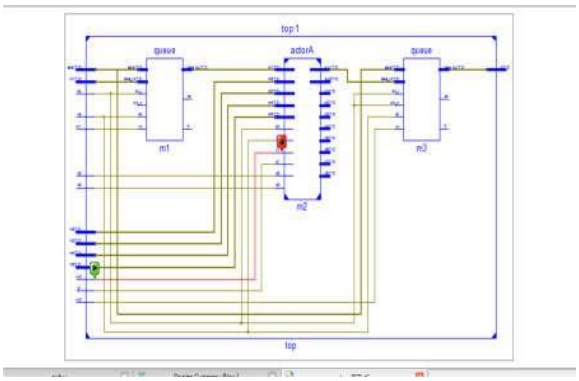

Fig.12: RTL schematic

The Technological schematic is shown in below Fig.13.

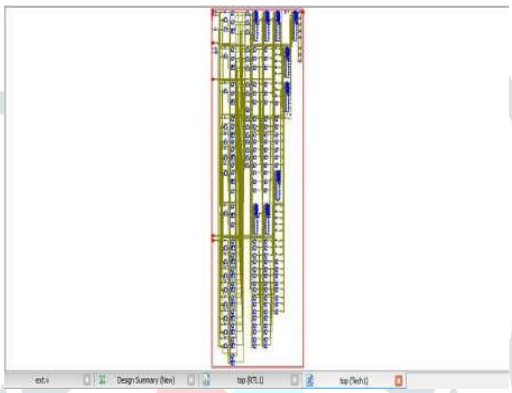

Fig.13: Technological schematic

The design and summary is shown in below Fig.14.

Device Utilization Summary			
Logic Utilization	Used	Available	Utilization
No of slices	134	3550	3%
No of Slice Flipflops	99	2367	2%
No of 4-input LUTs	95	1962	1%
No of bonded IOBs	45	135	50%
No of MULT8X8SIOs	1	10	60%
No of GCLKs	0	26	2%

Fig.14: Design and summary

The timing report is shown in below Fig.15.

```
  Clock period: 3.989ns (frequency: 250.617MHz)
  Total number of paths / destination ports: 3176 / 408
-------------------------------------------------------------------------
Delay:               3.989ns (Levels of Logic = 11)
  Source:            unit1/Mmult_w4_mult0001 (MULT)
  Destination:       m5/y1_7 (FF)
  Source Clock:      clk rising
  Destination Clock: clk rising

  Data Path: unit1/Mmult_w4_mult0001 to m5/y1_7
                                Gate     Net
    Cell:in->out      fanout   Delay   Delay  Logical Name (Net Name)
    ----------------------------------------  ------------
     MULT18X18SIO:CLK->P0     1    2.501    0.279  unit1/Mmult_w4_mult0001 (unit1/w4<0>)
     LUT2:I1->O               1    1.209    0.000  unit1/Madd_y_lut<0> (unit1/Madd_y_lut<0>)
     MUXCY:S->O               1    0.000    0.000  unit1/Madd_y_cy<0> (unit1/Madd_y_cy<0>)
     MUXCY:CI->O              1    0.000    0.000  unit1/Madd_y_cy<1> (unit1/Madd_y_cy<1>)
     MUXCY:CI->O              1    0.000    0.000  unit1/Madd_y_cy<2> (unit1/Madd_y_cy<2>)
     MUXCY:CI->O              1    0.000    0.000  unit1/Madd_y_cy<3> (unit1/Madd_y_cy<3>)
     MUXCY:CI->O              1    0.000    0.000  unit1/Madd_y_cy<4> (unit1/Madd_y_cy<4>)
     MUXCY:CI->O              1    0.000    0.000  unit1/Madd_y_cy<5> (unit1/Madd_y_cy<5>)
     XORCY:CI->O              2    0.000    0.622  unit1/Madd_y_xor<6> (w5<6>)
     LUT2:I0->O               1    0.000    0.000  Madd_w9_lut<6> (Madd_w9_lut<6>)
     MUXCY:S->O               0    0.000    0.000  Madd_w9_cy<6> (Madd_w9_cy<6>)
     XORCY:CI->O              1    0.000    0.000  Madd_w9_xor<7> (w9<7>)
     FDR:D                         0.000           m5/y1_7
    ----------------------------------------
    Total                          3.989ns (3.710ns logic, 0.279ns route)
```

Fig.15: Timing report

Comparison table of proposed work with existing work

Table 1: comparison table of proposed work with existing work

Parameters	Using clock gating technique	Clock gating technique using double edged triggered flip-flop
Delay	9.585ns	3.989ns
Frequency	104.330MHz	250.617MHz
Area	150/1962	95/1962

V. CONCLUSION

This paper provides a CG approach for dataflow models that can be automatically included in a HLS development flow's formulation phase. During the synthesis phase, the CG logic is produced along with the synthesis of the computing kernels linked to the dataflow network via FIFO queues. Experimental findings indicate that energy dissipation gains were accomplished with a small rise in the control logic without any decrease in the output.CG is not surprisingly appealing in circumstances where the structure is not fully utilized. As a consequence, this method is especially important in apps with dynamically variable output demands when it is difficult to design to a specific output level and when energy usage is considered expensive. Furthermore, CG inquiries should find more hostile control logic, whereby each actor is controlled, enabling higher freedom to actor inactivity.

REFERENCES

[1] S.Casale-Brunet, "Analysis and optimization of dynamic dataflow programs,"Ph.D. dissertation, STI Elect. Eng., STI, Lausanne, Switzerland, 2015.

[2] E. Bezati, "High-level synthesis of dataflow programs for heterogeneous platforms," Ph.D. dissertation, STI, Lausanne, Switzerland, 2015.

[3] S. C. Brunet et al., "Partitioning and optimization of high level stream applications for multi clock domain architectures," in Proc. IEEE Workshop Signal Process. Syst. (SiPS), Taipei, Taiwan, Oct. 2013, pp. 177–182.

[4] J. P. Oliver, J. Curto, D. Bouvier, M. Ramos, and E. Boemo, "Clock gating and clock enable for FPGA power reduction," in Proc. 8th Southern Conference on Programmable Logic (SPL), pp. 1-5, 2012.

[5] Warren Shum and Jason H. Anderson Department of Electrical and Computer Engineering, University of Toronto ―FPGA Glitch Power Analysis and Reduction, IEEE 2011.

[6] B. Ghavami and H. Pedram, "High performance asynchronous design flow using a novel static performance analysis method," Comput. Elect. *Eng.*, vol. 35, no. 6, pp. 920–941, Nov. 2009.

[7].S. Khawam et al, ―The reconfigurable instruction cell array,IEEE Trans. on VLSI Sys., vol. 16, no. 1, pp. 75-85, 2008.

[8]S. Casale-Brunet, M. Mattavelli, and J. W. Janneck, "Buffer optimization based on critical path analysis of a dataflow program design," in Proc. IEEE Int. Symp. Circuits Syst. (ISCAS) , Beijing, China, May 2013, pp. 1384–1387. [15] Analysis of Power Savings from Intelligent Clock Gating , Xilinx, San Jose, CA, USA, Aug. 2012.

[9] (2014). Open RVC-CAL Applications . Accessed on Feb. 25, 2014. [Online]. Available: http://github . com/orcc/orc-apps.

[10] M. Canale, S. Casale-Brunet, E. Bezati, M. Mattavelli, and J. Janneck, "Dataflow programs analysis and optimization using model predictive control techniques," J. Signal Process. Syst. , vol. 84, no. 3, pp. 371–381, 2015. [Online]. Available: http://dx . doi . org/10 . 1007/s11265-015-1083-4.

.

.