Apéndice A

Grafos de las aplicaciones usadas en la experimentación

Aquí las figuras de los grafos de la experimentación

A.0.4. Grafos que representan las estructuras obtenidas en la Técnica de Paralelización

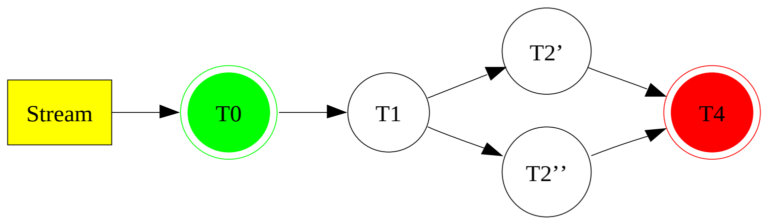


Figura A.1: Técnica de Paralelización: Grafo de tareas para el caso x2.

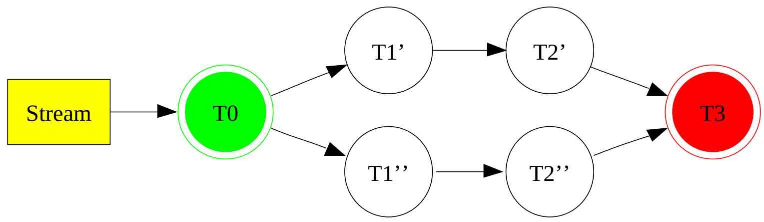


Figura A.2: Técnica de Paralelización: Grafo de tareas para el caso x4.

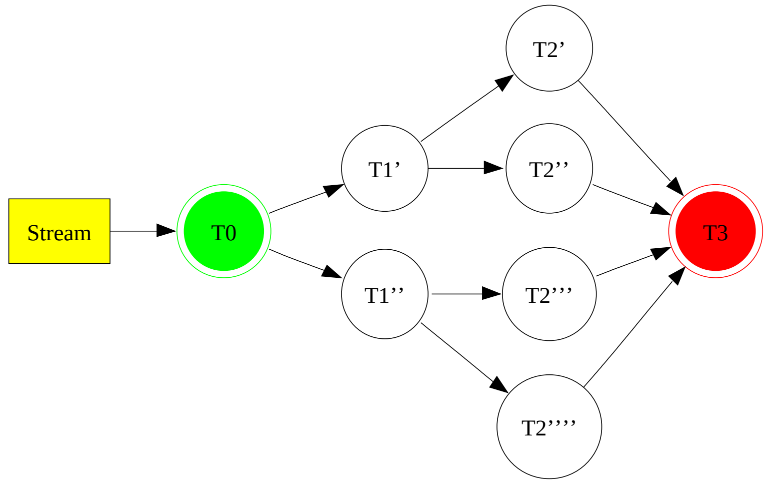


Figura A.3: Técnica de Paralelización: Grafo de tareas para el caso x8.

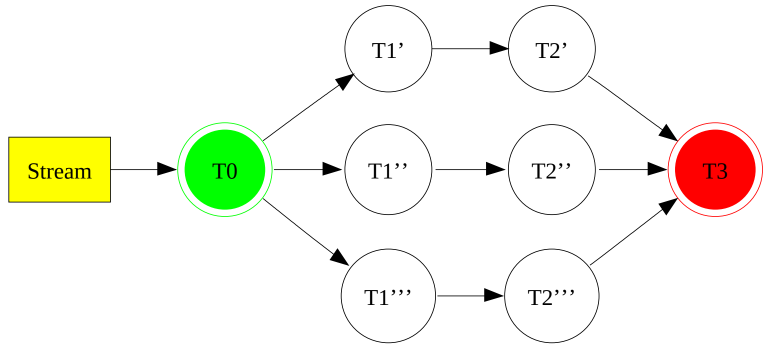


Figura A.4: Técnica de Paralelización: Grafo de tareas para el caso x16.

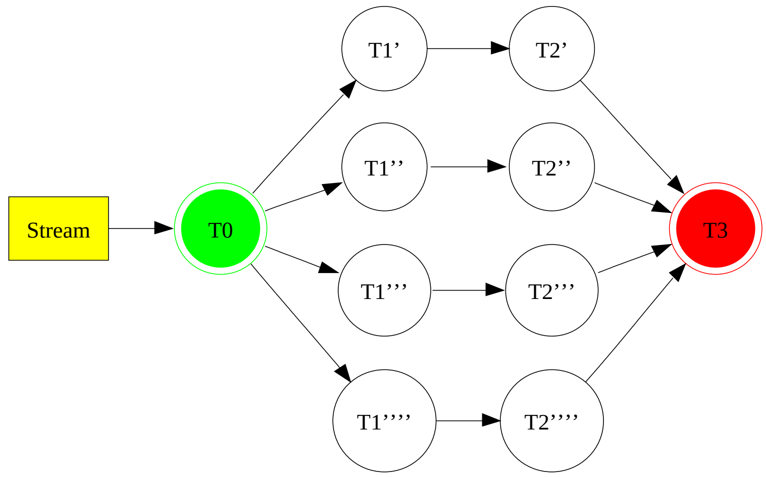


Figura A.5: Técnica de Paralelización: Grafo de tareas para el caso x32.

A.0.5. Grafos que representan las estructuras obtenidas en la Técnica de Replicación

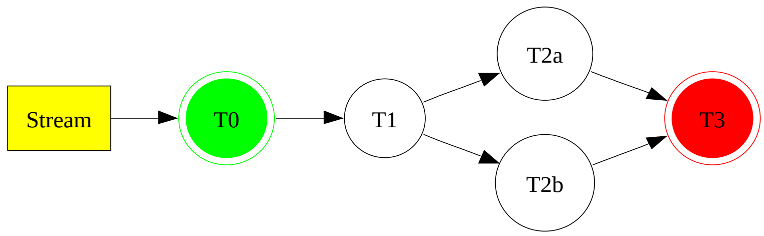


Figura A.6: Técnica de Replicación: Grafo de tareas para el caso x2.

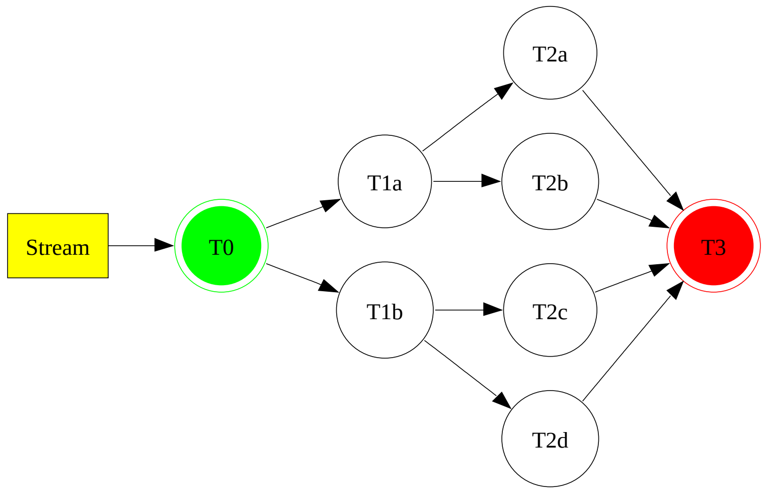


Figura A.7: Técnica de Replicación: Grafo de tareas para el caso x4.

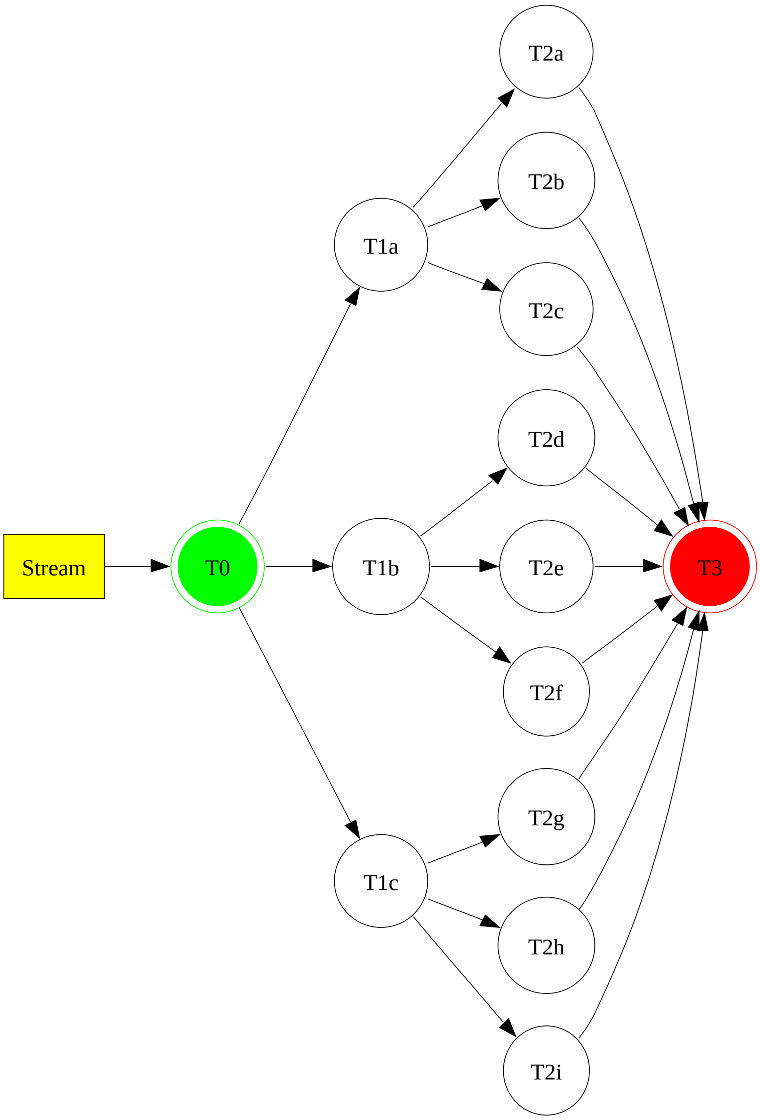
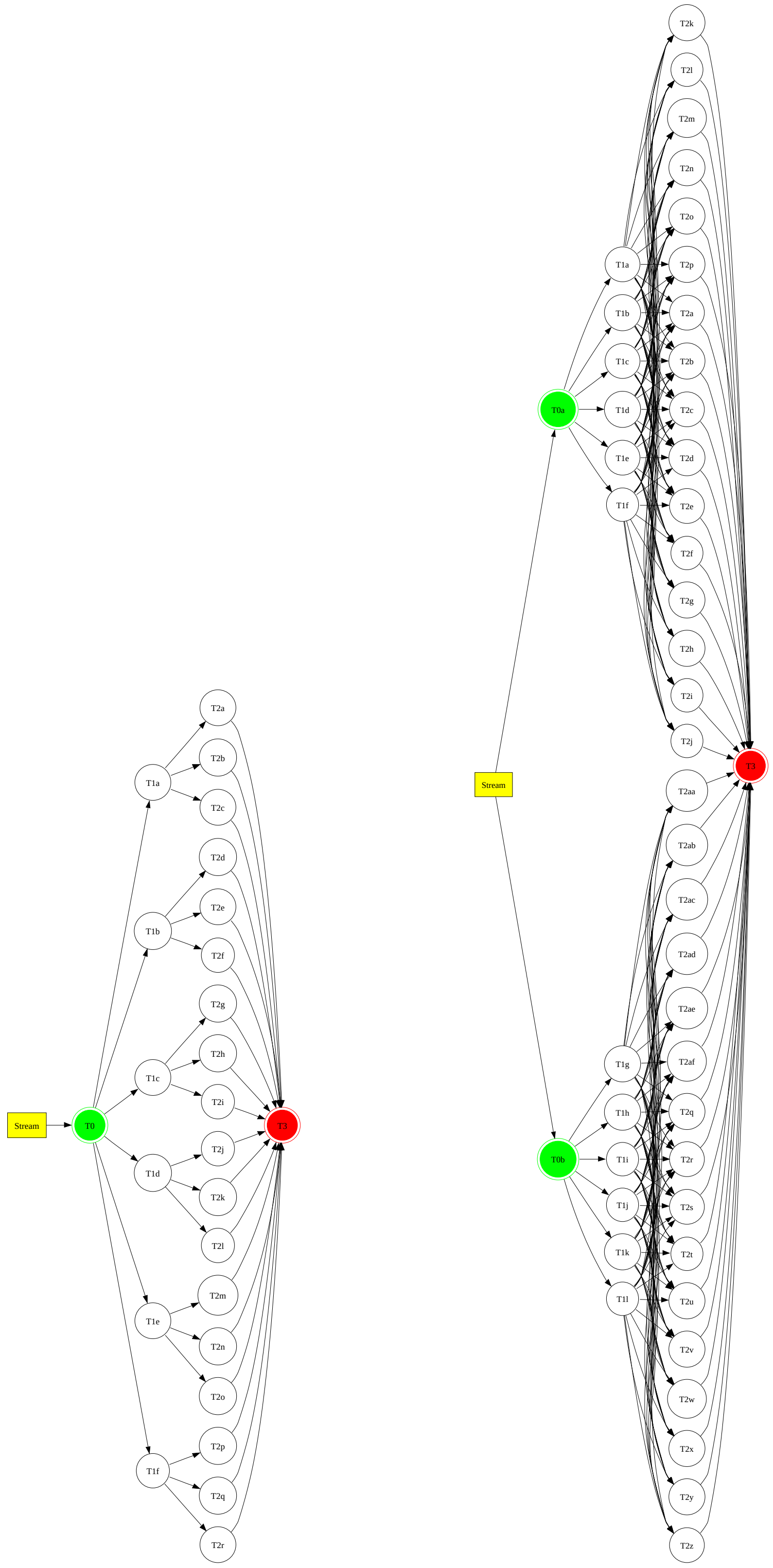


Figura A.8: Técnica de Replicación: Grafo de tareas para el caso x8.



(a) (b)

Cuadro A.1: Técnica de Replicación: Grafo de tareas para el caso (a) x16 y (b) x32.

A.0.6. Grafos de las aplicaciones homogéneas y arbitrarias

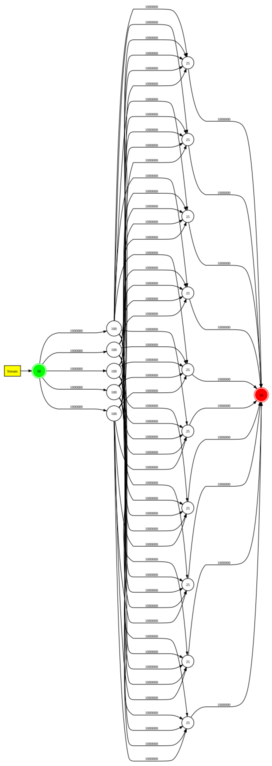


Figura A.9: Grafo de tareas de la aplicación ho01.

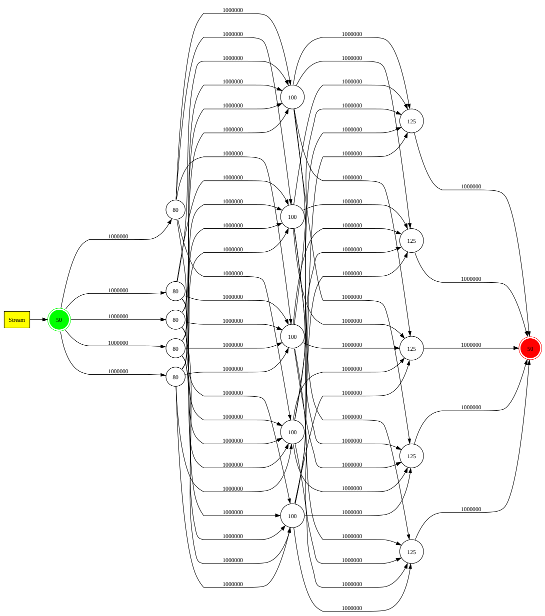


Figura A.10: Grafo de tareas de la aplicación ho02.

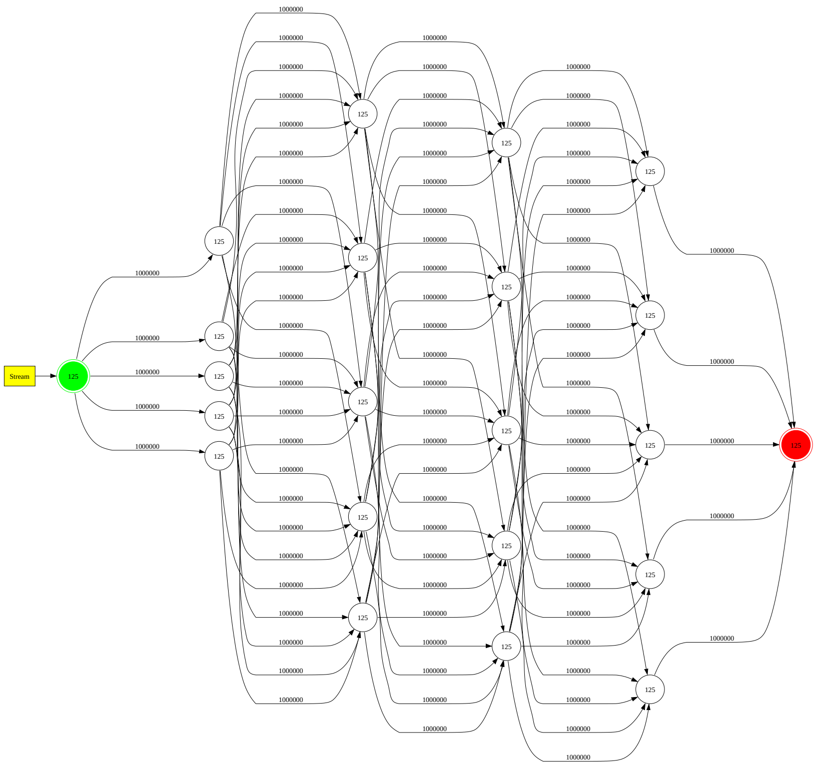


Figura A.11: Grafo de tareas de la aplicación ho03.

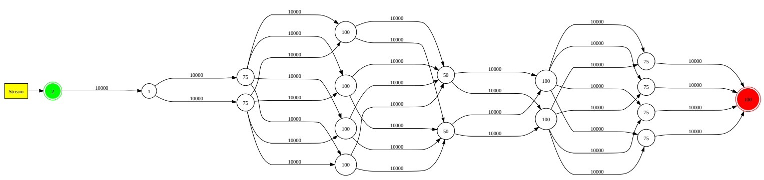


Figura A.12: Grafo de tareas de la aplicación ho03.

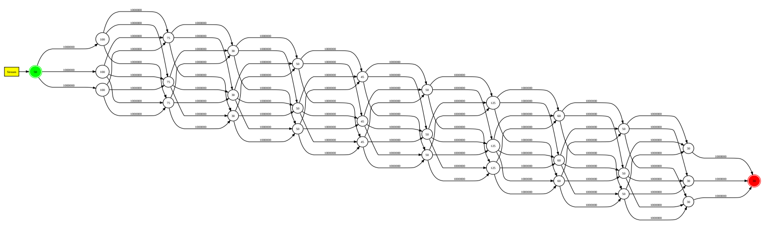


Figura A.13: Grafo de tareas de la aplicación ho05.

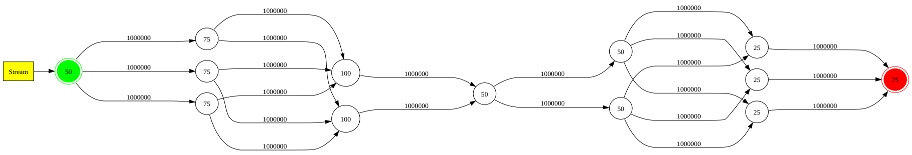


Figura A.14: Grafo de tareas de la aplicación ho06.

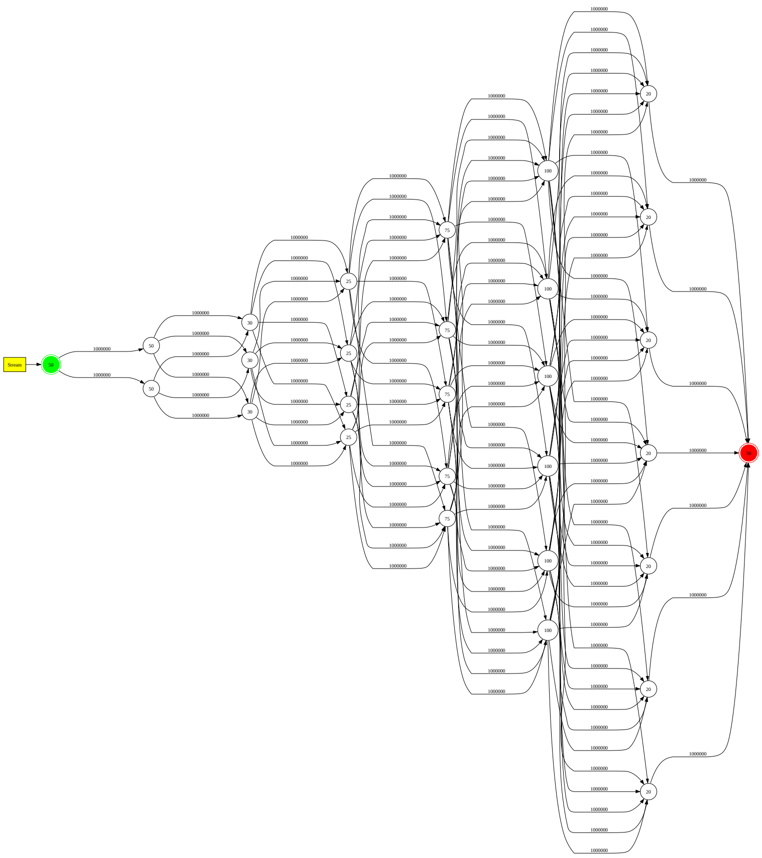


Figura A.15: Grafo de tareas de la aplicación ho07.

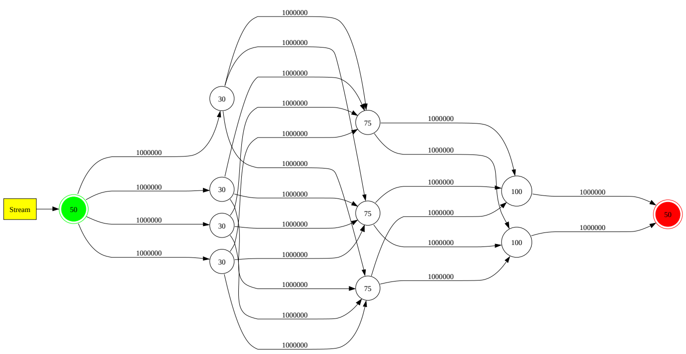


Figura A.16: Grafo de tareas de la aplicación ho08.

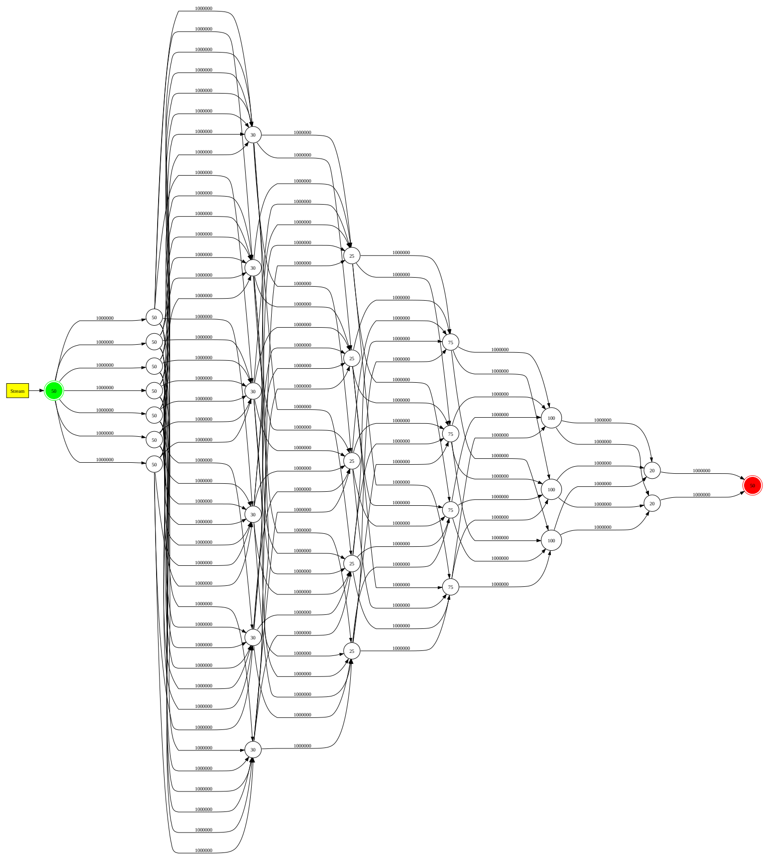


Figura A.17: Grafo de tareas de la aplicación ho09.

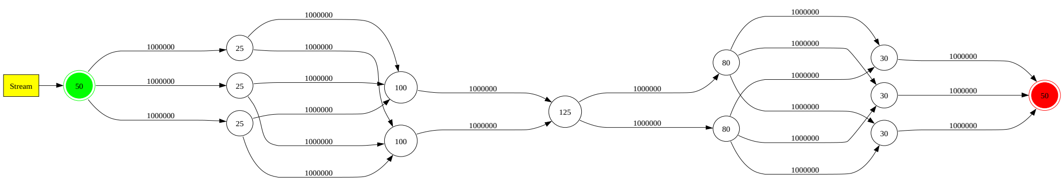


Figura A.18: Grafo de tareas de la aplicación ho10.

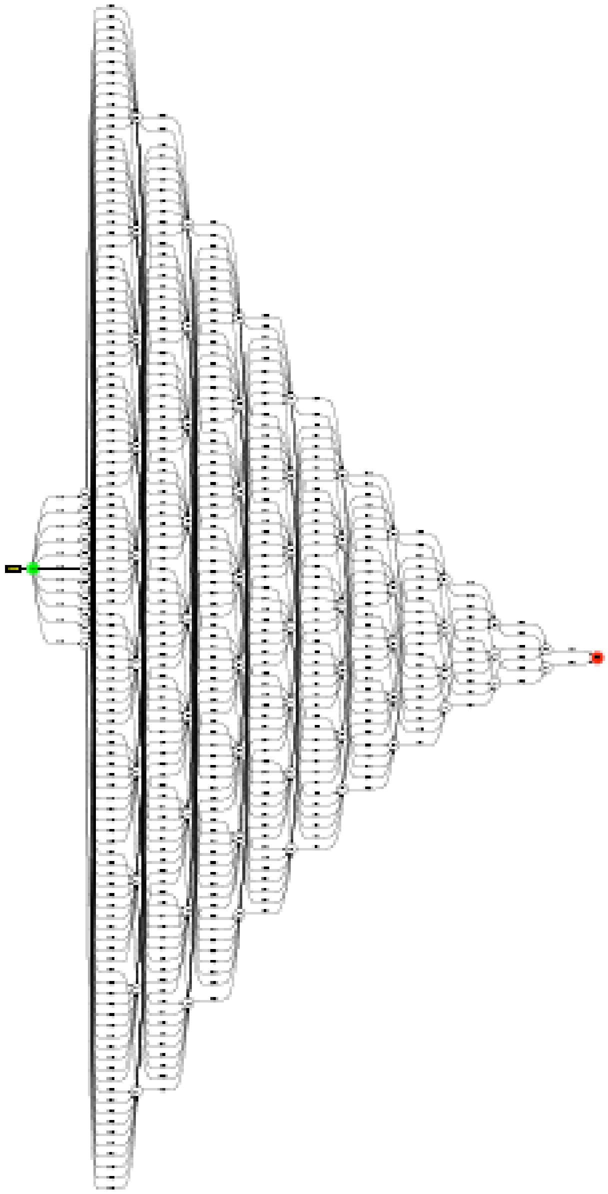


Figura A.19: Grafo de tareas de la aplicación ar01.

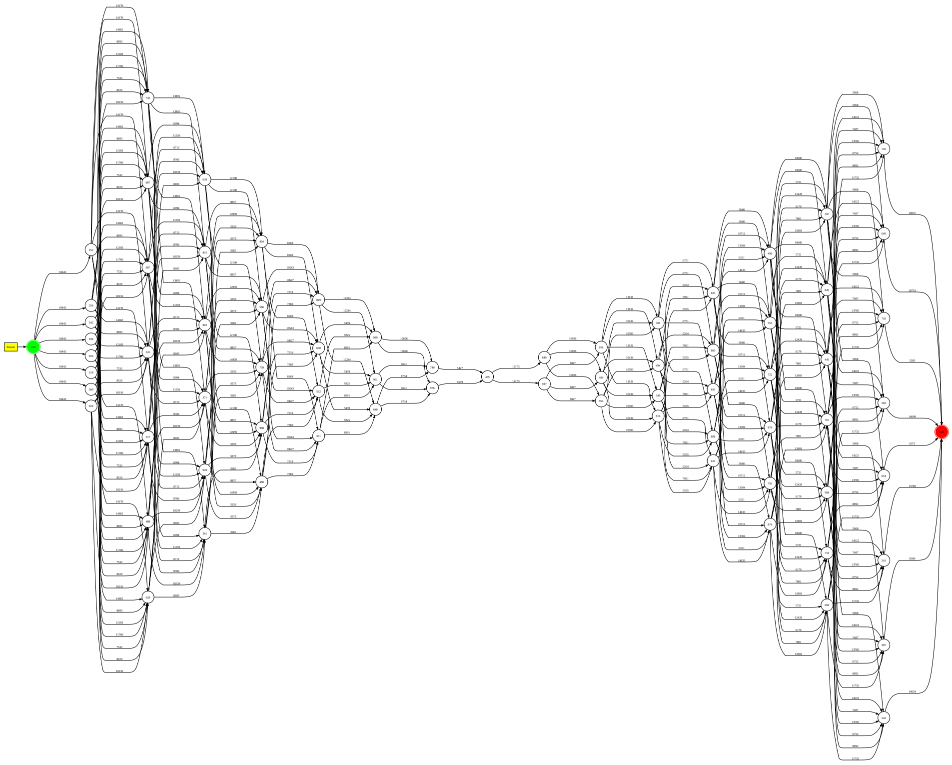


Figura A.20: Grafo de tareas de la aplicación ar02.

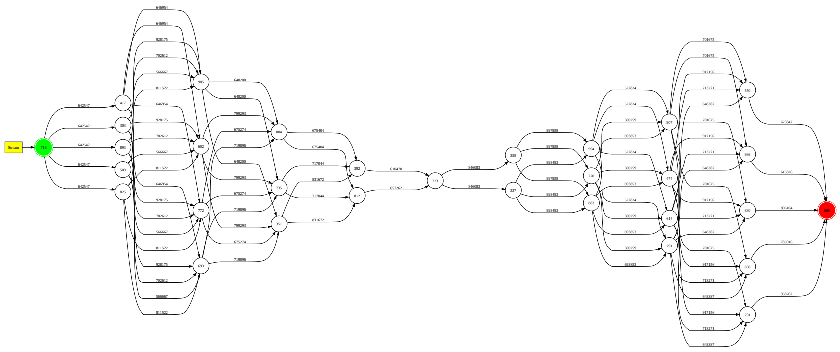


Figura A.21: Grafo de tareas de la aplicación ar03.

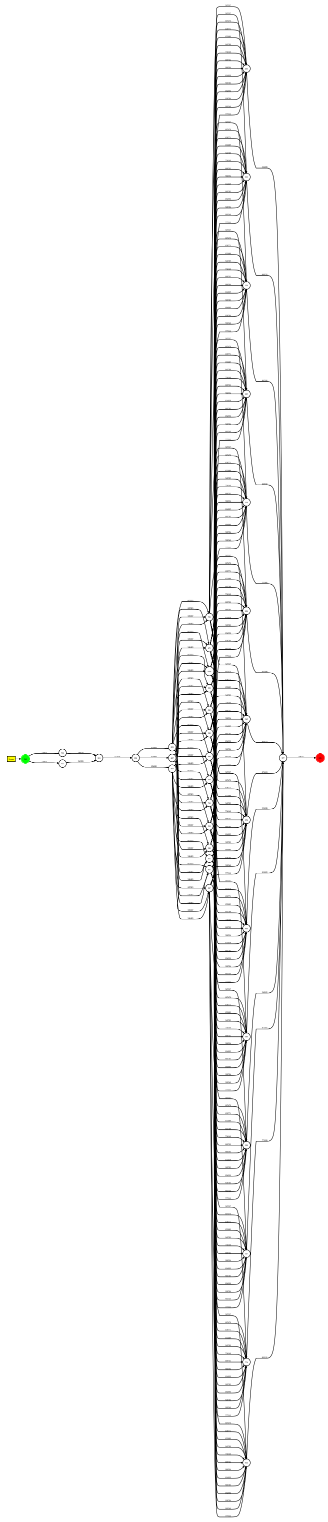


Figura A.22: Grafo de tareas de la aplicación ar04.

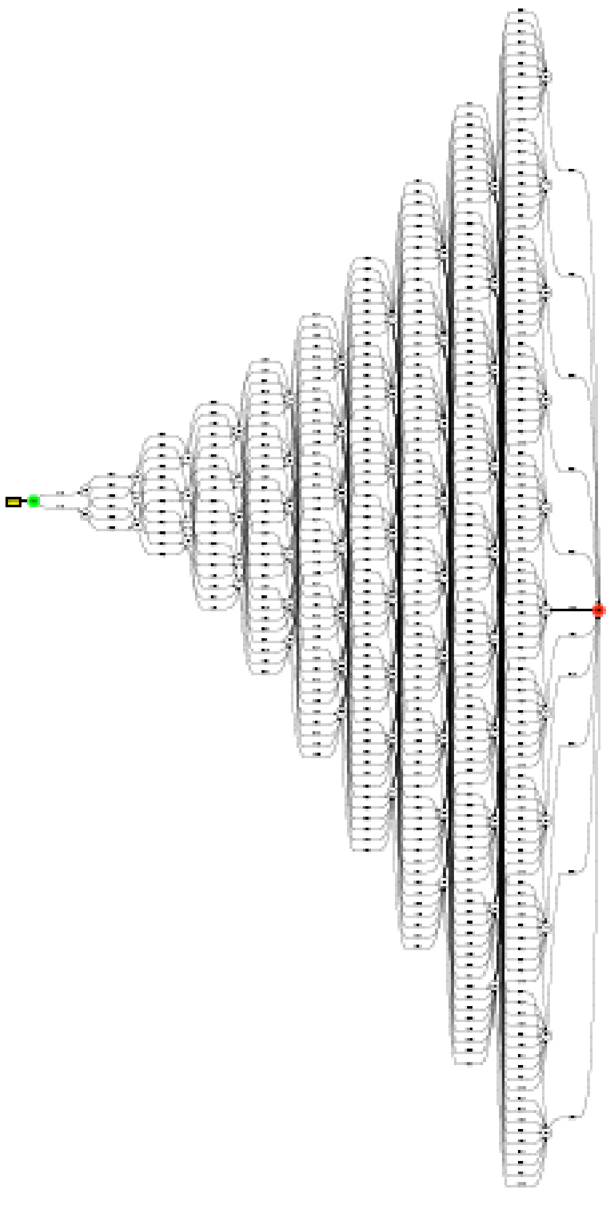


Figura A.23: Grafo de tareas de la aplicación ar05.

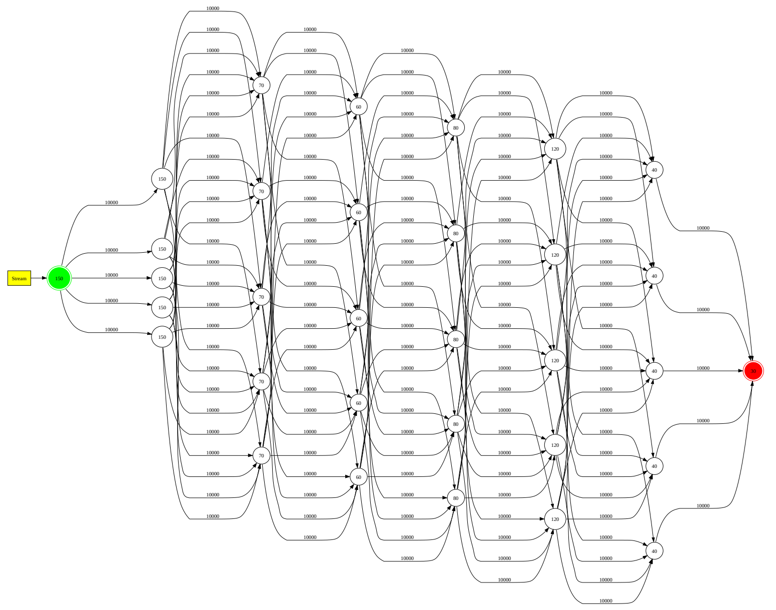


Figura A.24: Grafo de tareas de la aplicación ar06.

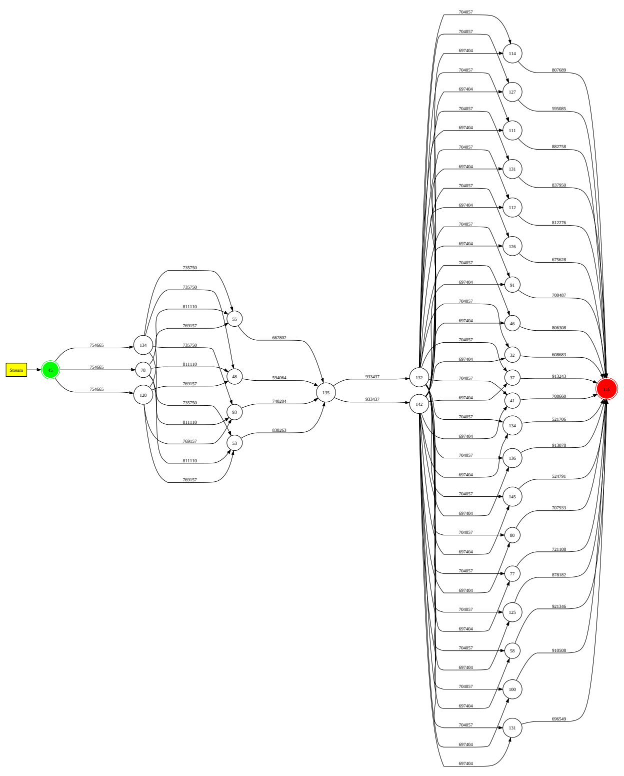


Figura A.25: Grafo de tareas de la aplicación ar07.

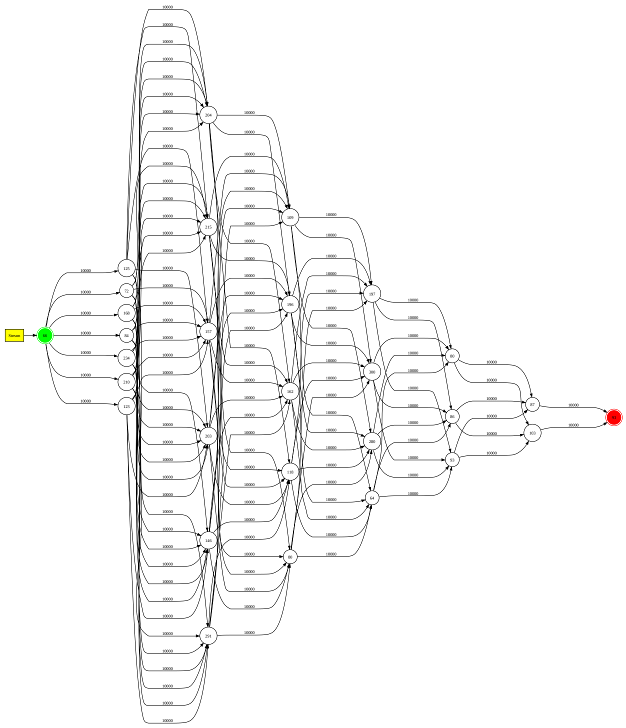


Figura A.26: Grafo de tareas de la aplicación ar08.

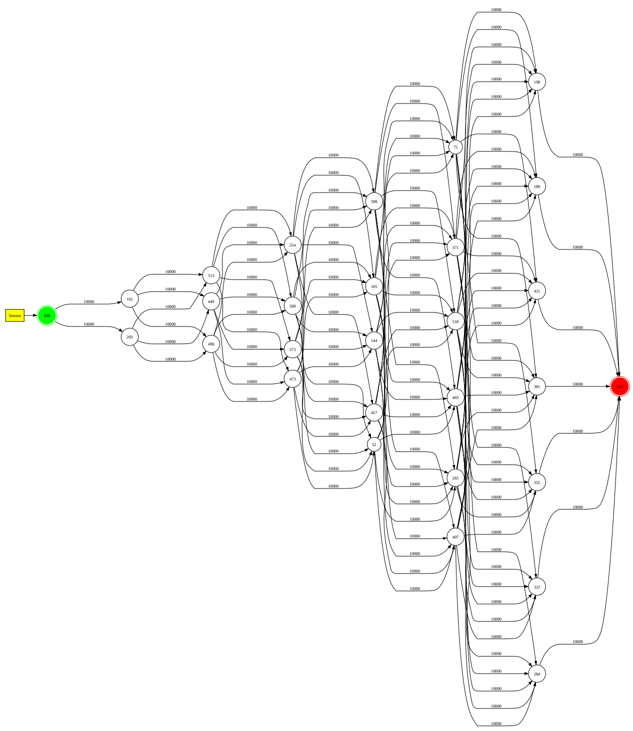


Figura A.27: Grafo de tareas de la aplicación ar09.

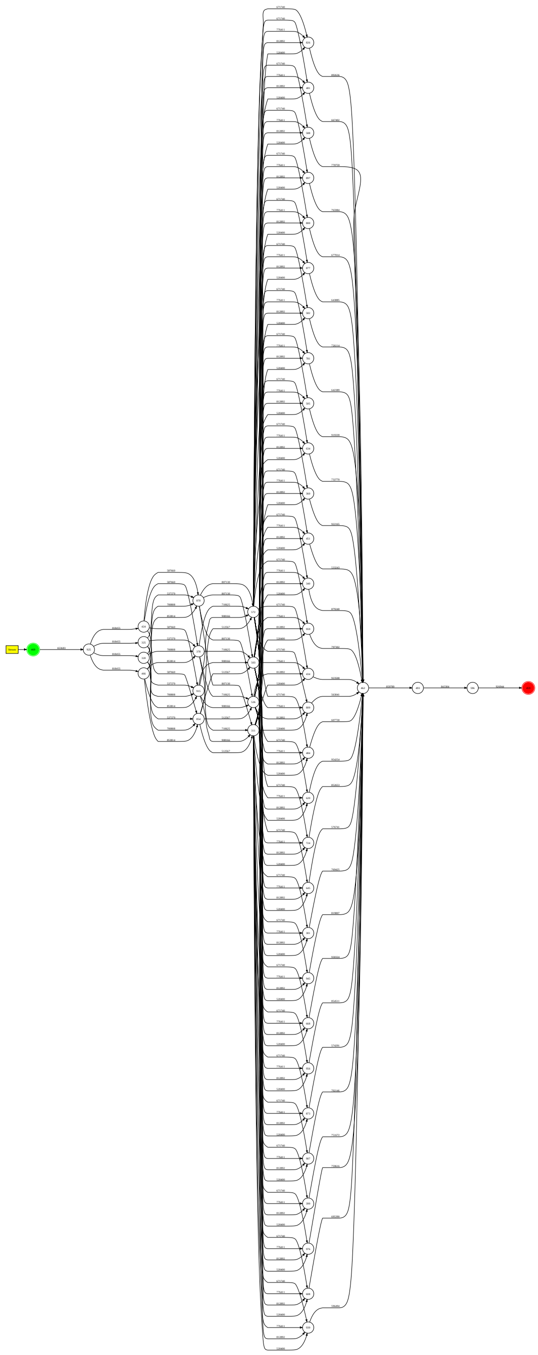


Figura A.28: Grafo de tareas de la aplicación ar10.

A.0.7. Grafos obtenidos para el compresor MPEG2

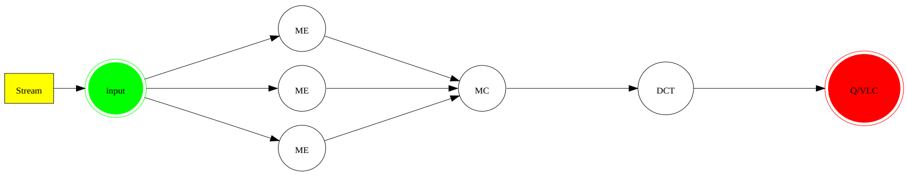


Figura A.29: MPEG2: Grafo de tareas de la aplicación para el caso x2.

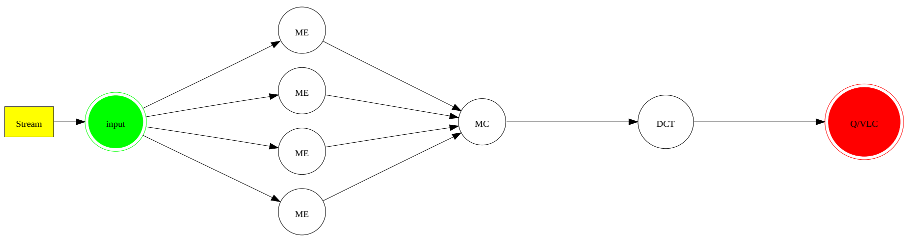


Figura A.30: MPEG2: Grafo de tareas de la aplicación para el caso x4.

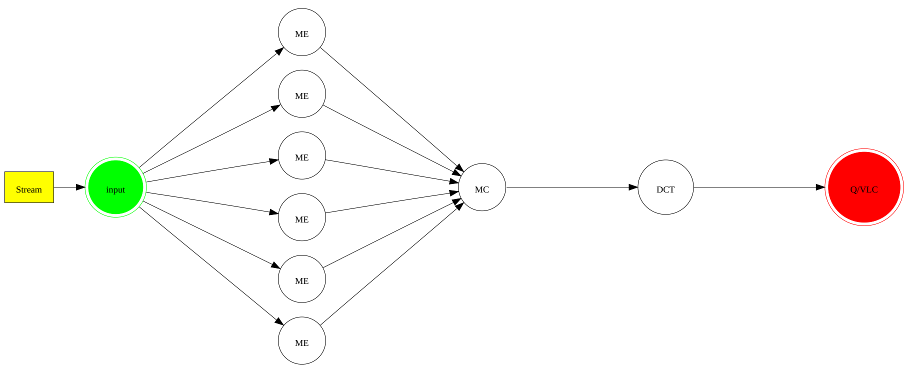


Figura A.31: MPEG2: Grafo de tareas de la aplicación para el caso x8.

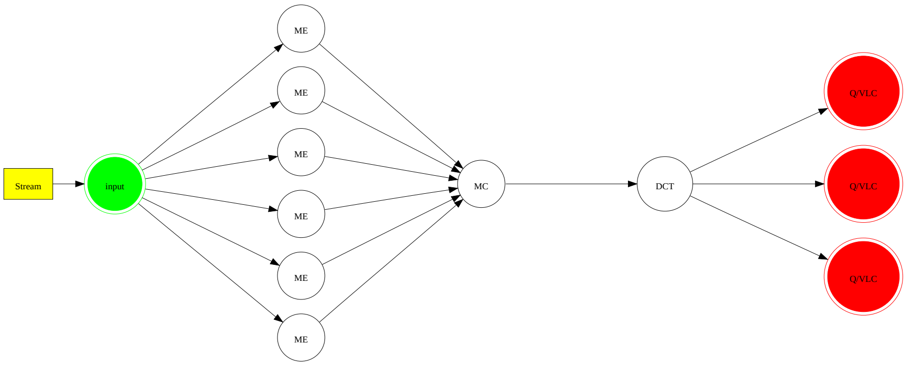


Figura A.32: MPEG2: Grafo de tareas de la aplicación para el caso x16.

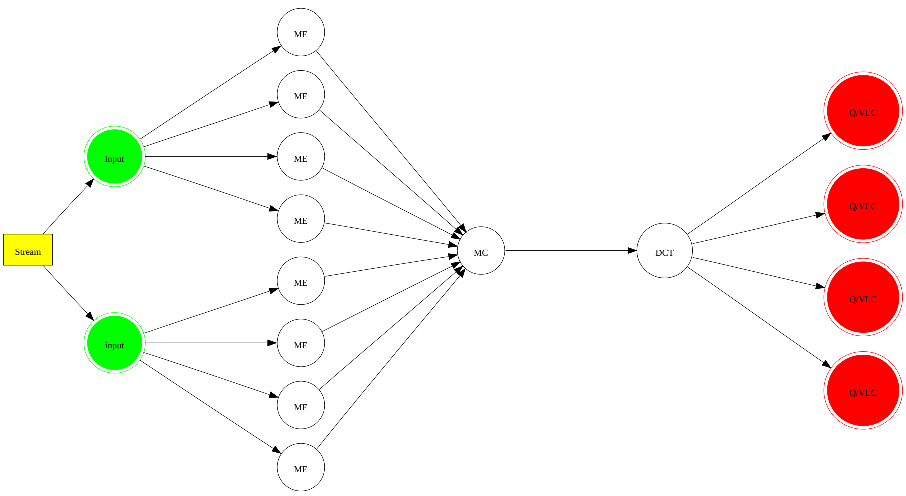


Figura A.33: MPEG2: Grafo de tareas de la aplicación para el caso x32.

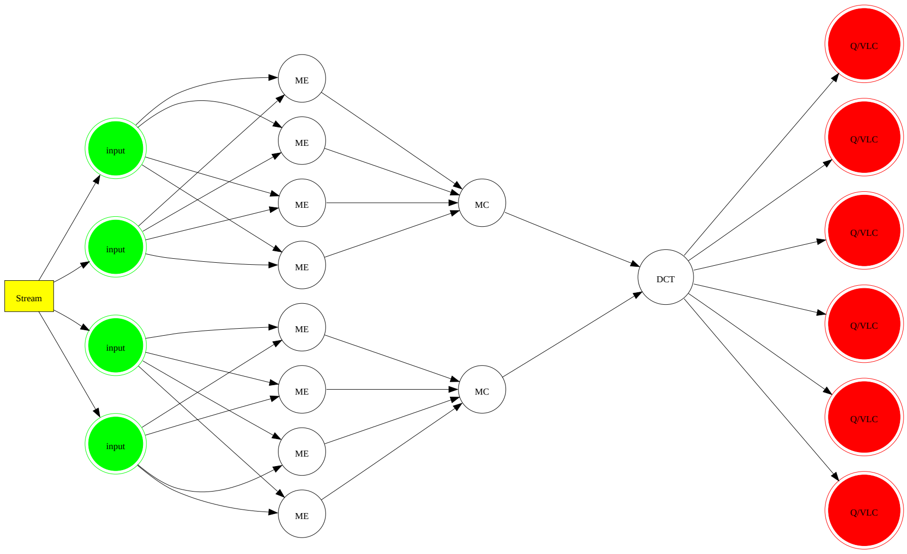


Figura A.34: MPEG2: Grafo de tareas de la aplicación para el caso x64.

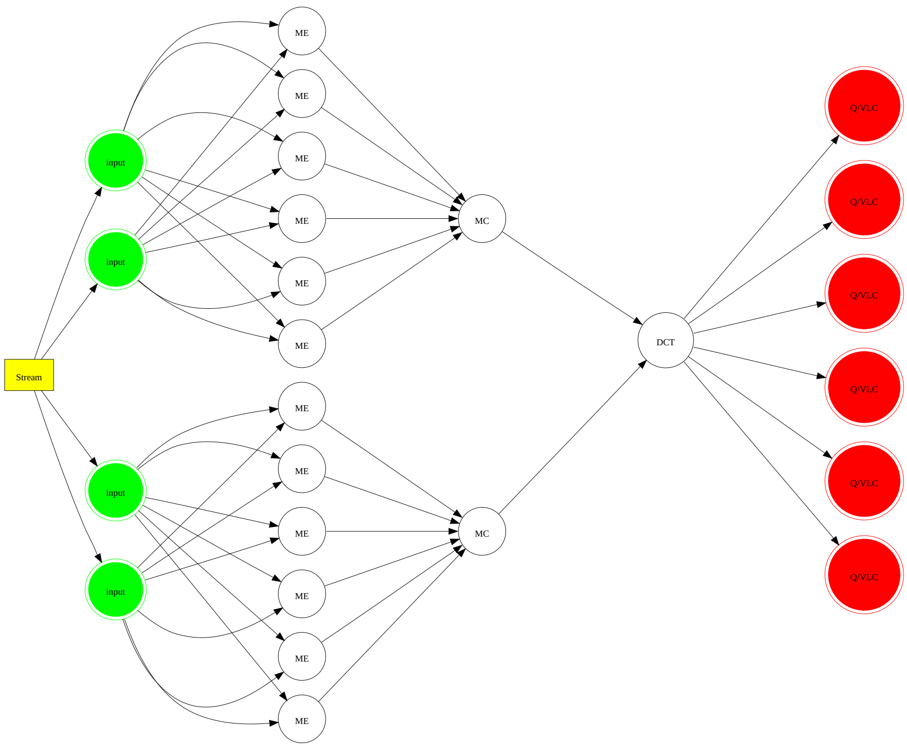


Figura A.35: MPEG2: Grafo de tareas de la aplicación para el caso x128.

A.0.8. Grafos obtenidos para la aplicación IVUS (*IntraVascular Ultrasound*)

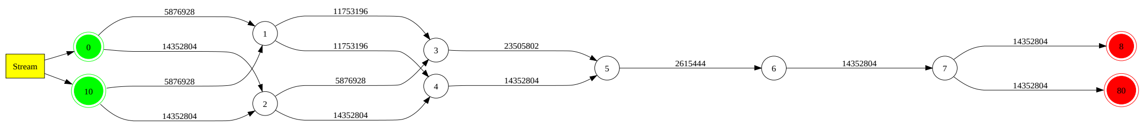


Figura A.36: IVUS: Grafo de tareas de la aplicación para el caso x2.

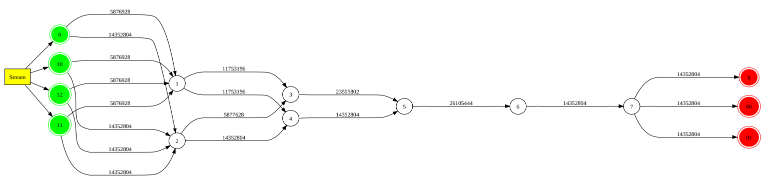


Figura A.37: IVUS: Grafo de tareas de la aplicación para el caso x4.

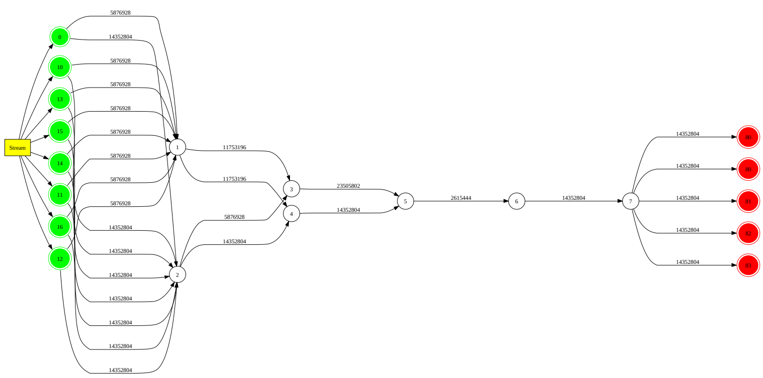


Figura A.38: IVUS: Grafo de tareas de la aplicación para el caso x8.

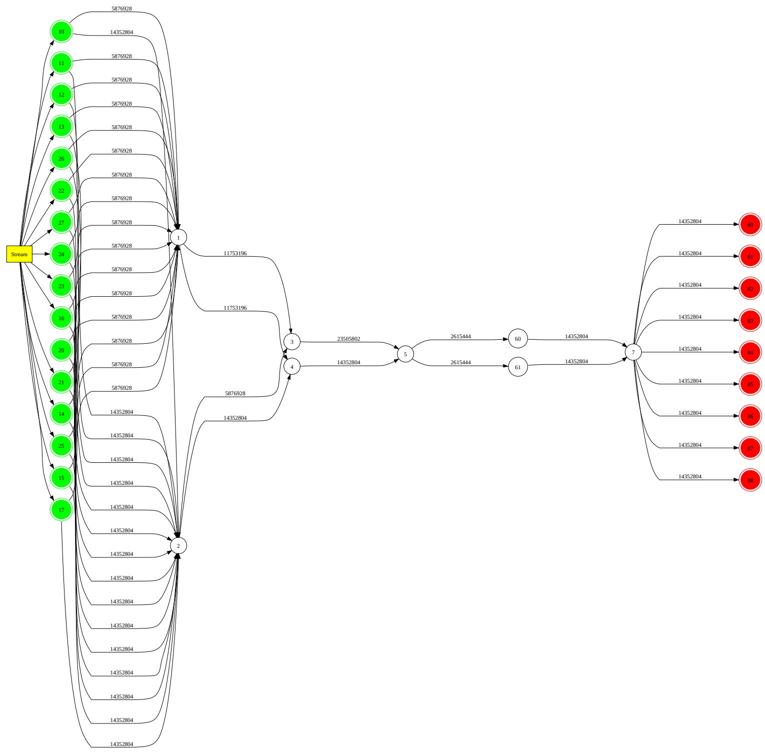


Figura A.39: IVUS: Grafo de tareas de la aplicación para el caso x16.

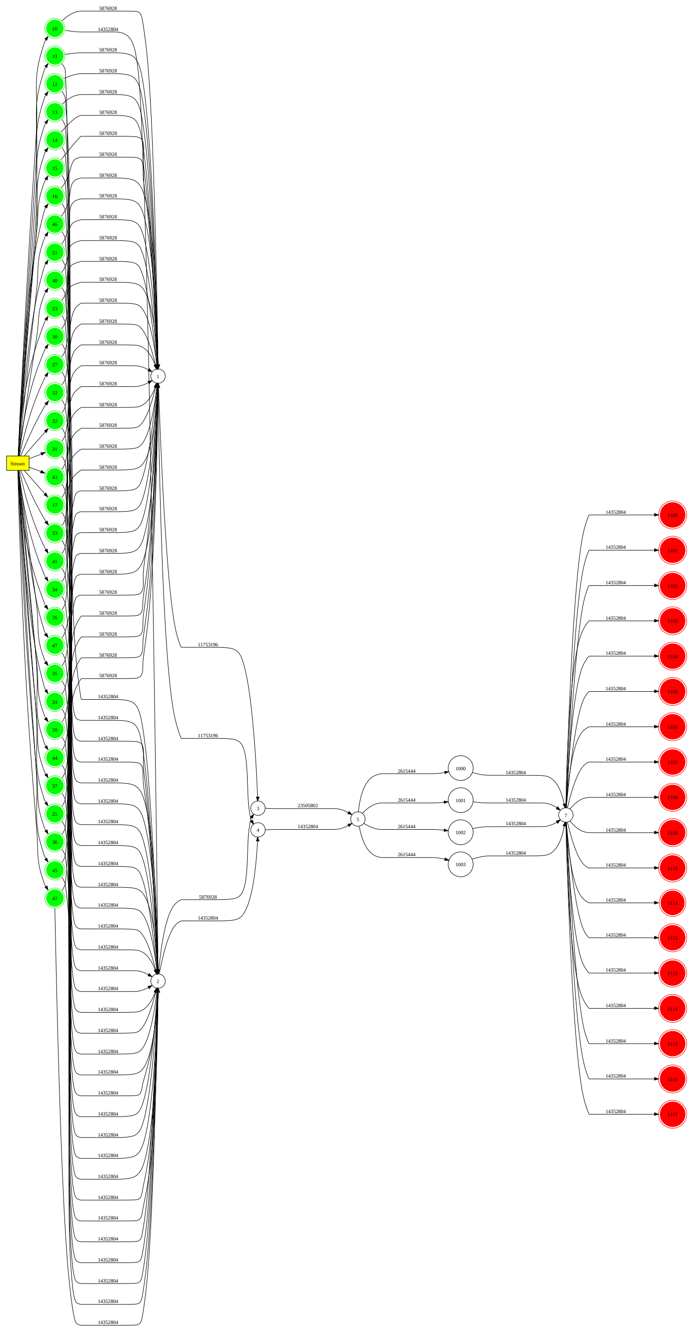
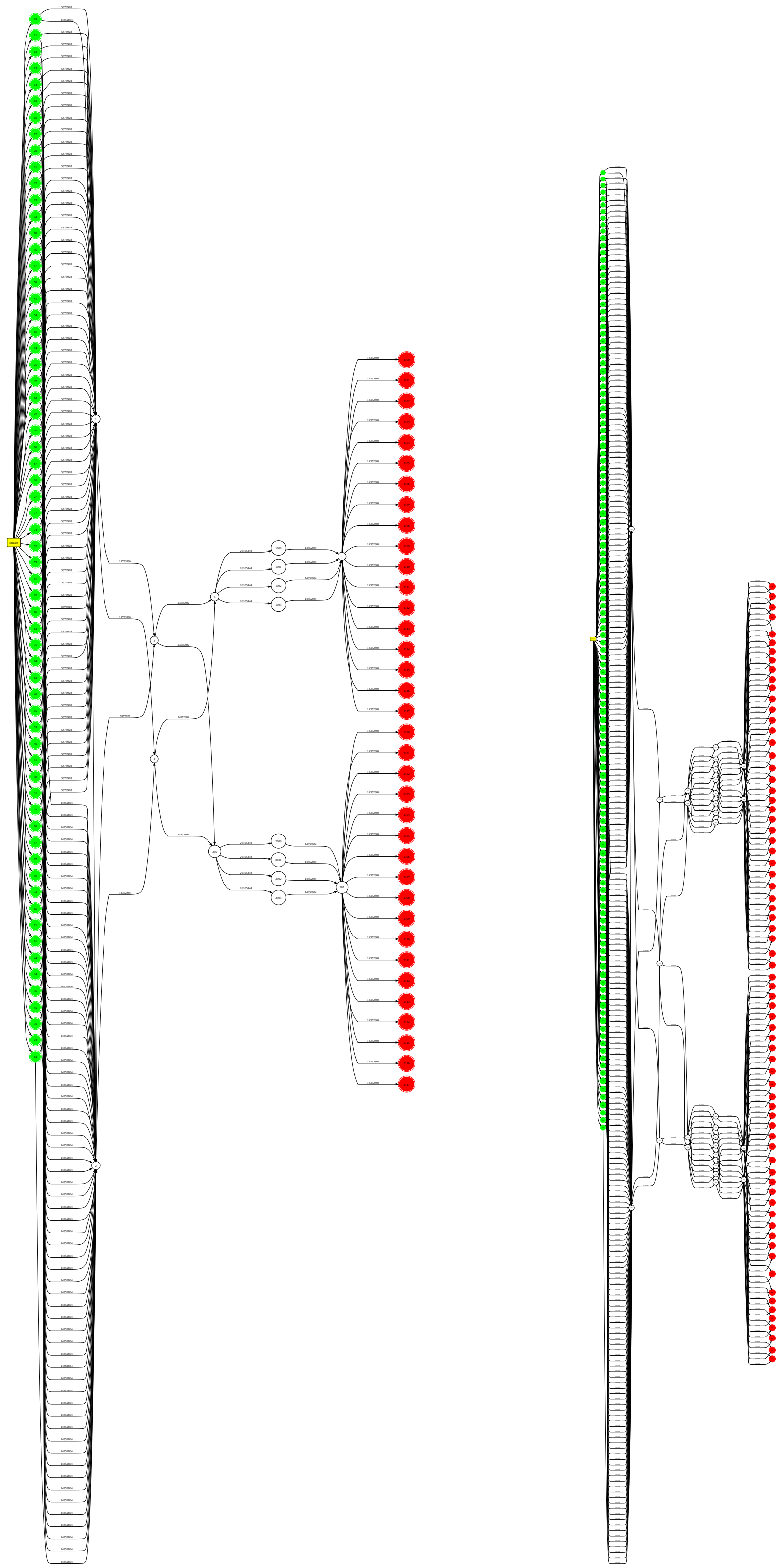


Figura A.40: IVUS: Grafo de tareas de la aplicación para el caso x32.



(a) (b)

Cuadro A.2: IVUS: Grafo de tareas de la aplicación para el caso (a) x64 y (b) x128.

A.0.9. Grafos obtenidos para la aplicación BASIZ (*Bright and Saturated Image Zones*)

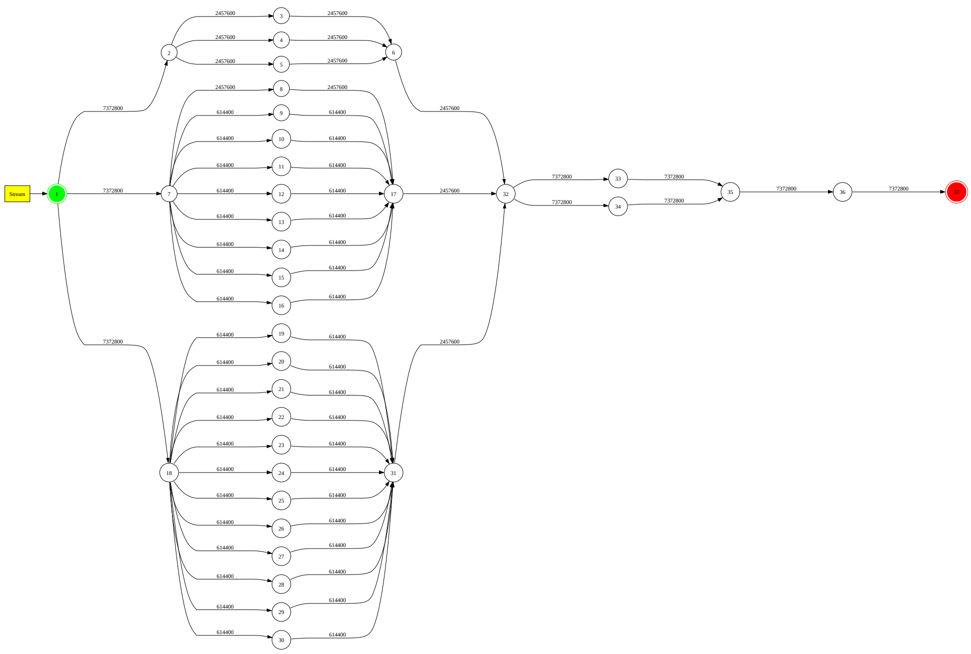


Figura A.41: BASIZ: Grafo de tareas de la aplicación para el caso x2.

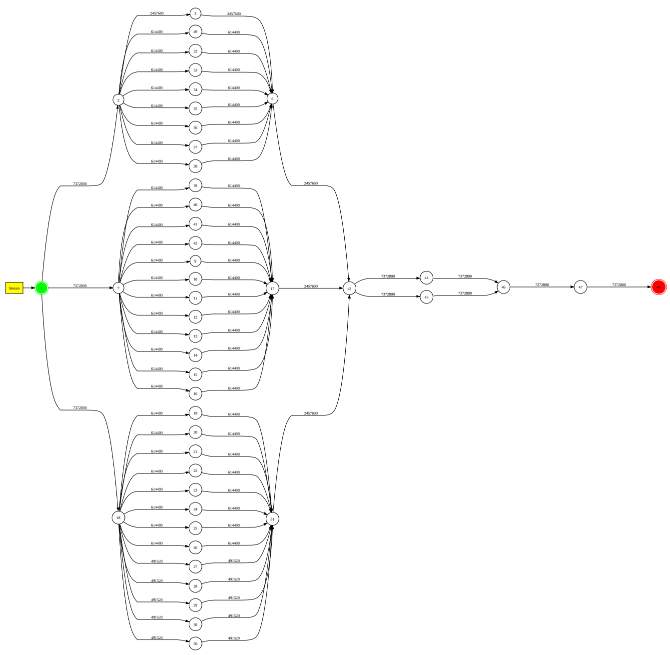


Figura A.42: BASIZ: Grafo de tareas de la aplicación para el caso x4.

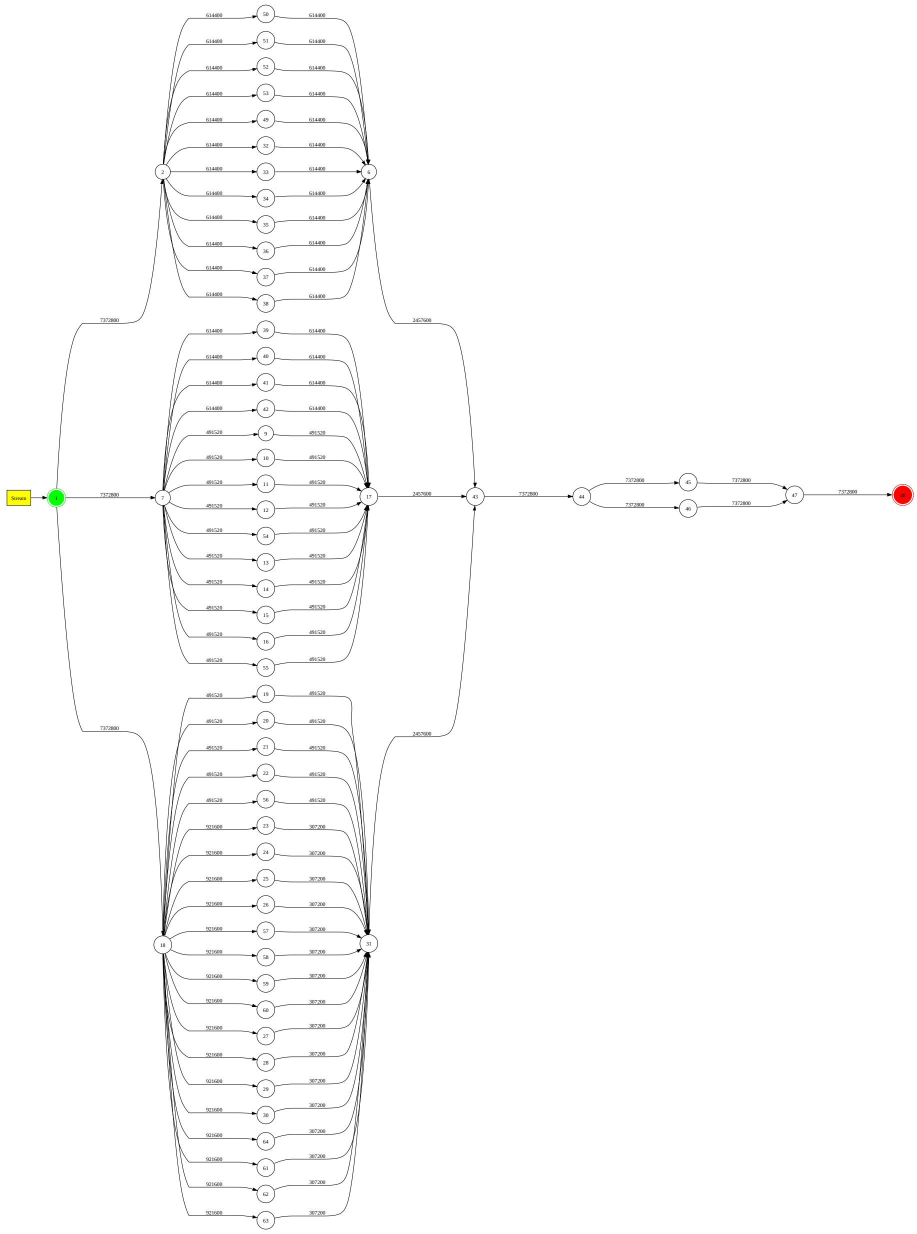
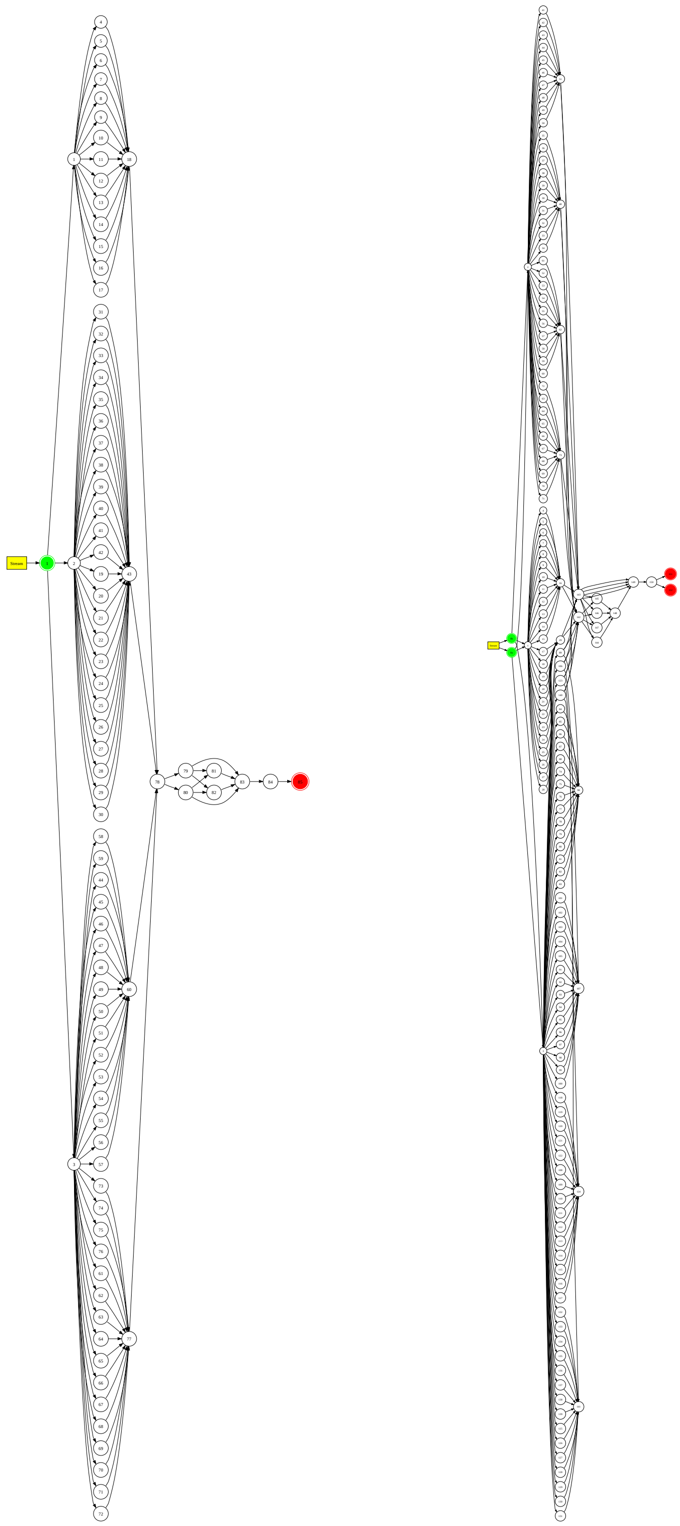


Figura A.43: BASIZ: Grafo de tareas de la aplicación para el caso x8.



(a) (b)

Cuadro A.3: BASIZ: Grafo de tareas de la aplicación para el caso (a) x16 y (b) x32.

Bibliografía

[AA04] Nitin Auluck and Dharma P. Agrawal. An integrated scheduling algorithm for precedence constrained hard and soft real-time tasks on heterogeneous multiprocessors. In *Embedded and Ubiquitous Computing*, volume 3207 of *LNCS*, pages 196–206, 2004.

[AAL95] Shahriar M. Akramullah, Ishfaq Ahmad, and Ming L. Liou. A data-parallel approach for real-time MPEG-2 video encoding. *Journal on Parallel and Distributed Compututing*, 30(2):129–146, 1995.

[AHL02] Ishfaq Ahmad, Yong He, and Ming L. Liou. Video compression with parallel processing. *Parallel Computing*, 28(7-8):1039–1078, 2002.

[AISS97] Albert Alexandrov, Mihai F. Ionescu, Klaus E. Schauser, and Chris Scheiman. LogGP: Incorporating long messages into the LogP model for parallel computation. *Journal of Parallel and Distributed Computing*, 44(1):71–79, 1997.

[CA06] Liang Chen and Gagan Agrawal. Supporting self-adaptation in streaming data mining applications. In *IEEE Proceedings on Parallel and Distributed Processing Symposium (IPDPS'06)*, pages CD–ROM, 2006.

[CB04] Anthony J. Cox and Clive G. Brown. Massively parallel DNA sequencing using single molecule array technology. In *12th Intelligent Systems for Molecular Biology (ISMB'04)*, august 2004.

[CGS07] Kirk W. Cameron, Rong Ge, and Xian-He Sun. log_nP and log_3P: Accurate analytical models of point-to-point communication in distributed systems. *IEEE Transactions on Computers*, 56(3):314–327, 2007.

[CkLW$^+$00] Alok Choudhary, Wei keng Liao, Donald Weinerr, Pramod Varshwey, Richard Linderman, and Mark Linderman. Design implementation and evaluation of parallel pipelined STAP on parallel computers. *IEEE Transactions on Aerospace and Electronics Systems*, 36(2):528–548, April 2000.

[CKP$^+$93] David E. Culler, Richard M. Karp, David A. Patterson, Abhijit Sahay, Klaus E. Schauser, Eunice Santos, Ramesh Subramonian, and Thorsten von Eicken. LogP: Towards a realistic model of parallel computation. In *Principles Practice of Parallel Programming*, pages 1–12, 1993.

[CNNS94] Alok Choudhary, Bhagirath Narahari, David Nicol, and Rahul Simha. Optimal processor assignment for a class of pipelined computations. *IEEE Transactions on Parallel and Distributed Systems*, 5(4):439–445, 1994.

[CSL05] Eduardo César, Joan Sorribes, and Emilio Luque. Modeling pipeline applications in POETRIES. In *Euro-Par 2005 Parallel Processing*, volume 3648 of *LNCS*, pages 83–92, 2005.

[DA04] Wei Du and Gagan Agrawal. Packet size optimization for supporting coarse-grained pipelined parallelism. In *Proceedings of the 2004 International Conference on Parallel Processing (ICPP'04)*, pages 259–266, 2004.

[DA05] Wei Du and Gagan Agrawal. Filter decomposition for supporting coarse-grained pipelined parallelism. In *Proceedings of the 2005 International Conference on Parallel Processing (ICPP'05)*, pages 539–546, 2005.

[DFA03] Wei Du, Renato Ferreira, and Gagan Agrawal. Compiler support for exploiting coarse-grained pipelined parallelism. In *IEEE Proceedings of the International Conference on Supercomputing (ISC '03)*, pages CD–ROM, 2003.

[DSAW99] Christian Demant, Bernd Streicher-Abel, and Peter Waszkewitz. *Industrial Image Processing Visual Quality Control in Manufacturing*. Springer-Verlag, 1999.

[FARB06] J. Fernández, Mancia Anguita, Eduardo Ros, and José Luis Bernier. SCE toolboxes for the development of high-level parallel applications. In *International Conference on Computational Science (2)*, volume 3992 of *LNCS*, pages 518–525, 2006.

[GHC+05] Debora Gil, Aura Hernandez, Antoni Carol, Oriol Rodriguez, and Petia Radeva. A deterministic-statistic adventitia detection in IVUS images. In *Functional Imaging and Modeling of the Heart (FIMH'05)*, volume 3504 of *LNCS*, pages 65–74, 2005.

[GHR+06] Debora Gil, Aura Hernandez, Oriol Rodriguez, Josepa Mauri, and Petia Radeva. Statistical strategy for anisotropic adventitia modelling in IVUS. *IEEE Transactions on Medical Imaging*, 25(6):768–778, 2006.

[Gro] MPEG Software SiMulation Group.

[GRR+03] Fernando Guirado, Ana Ripoll, Concepció Roig, Xiao Yuan, and Emilio Luque. Predicting the best mapping for efficient exploitation of task and data parallelism. In *Euro-Par 2003 Parallel Processing*, volume 2790 of *LNCS*, pages 218–223, 2003.

[GRR+06] Fernando Guirado, Ana Ripoll, Concepció Roig, Aura Hernández, and Emilio Luque. Exploiting throughput for pipeline execution in streaming image processing applications. In *Euro-Par 2006 Parallel Processing*, volume 4128 of *LNCS*, pages 1095–1105, 2006.

[GRRL04a] Fernando Guirado, Ana Ripoll, Concepció Roig, and Emilio Luque. Performance prediction using an application-oriented mapping tool. In *IEEE Proceedings of the 8th. Euromicro Workshop on Parallel, Distributed and Network-based Processing (PDP'04)*, pages 184–191, 2004.

[GRRL04b] Fernando Guirado, Ana Ripoll, Concepció Roig, and Emilio Luque. A pipeline-based approach for mapping message-passing applications with an input data stream. In *Recent Advances in Parallel Virtual Machine and Message Passing Interface (EuroPVM/MPI-2004)*, volume 3241 of *LNCS*, pages 224–233, 2004.

[GRRL05a] Fernando Guirado, Ana Ripoll, Concepció Roig, and Emilio Luque. Exploitation of parallelism for applications with an input data stream: Optimal resource-throughput tradeoffs. In *IEEE Proceedings of the 8th. Euromicro Workshop on Parallel, Distributed and Network-based Processing (PDP'05)*, pages 170–178, 2005.

[GRRL05b] Fernando Guirado, Ana Ripoll, Concepció Roig, and Emilio Luque. Optimizing latency under throughput requirements for streaming applications on cluster execution. In *IEEE International Conference on Cluster Computing (Cluster 2005)*, pages CD–ROM, 2005.

[GYK+00] Tarak Gandhi, Mau-Tsuen Yang, Rangachar Kasturi, Octavia I. Camps, Lee D. Coraor, and Jeffrey McCandless. Detection of obstacles in the flight path of an aircraft. In *Conference on Computer Vision and Pattern Recognition (CVPR'00)*, pages 2304–2311, October 2000.

[HCAL89] J. J. Hwang, Y. C. Chow, F. D. Anger, and C. Y. Lee. Scheduling precedence task graphs in systems with interprocessor communication times. *SIAM Journal on Computing*, 18(2):244–257, April 1989.

[HMM05] Kevin A. Huck, Allen D. Malony, and Alan Morris. Design and implementation of a parallel performance data management framework. In *Proceedings of the 2005 International Conference on Parallel Processing (ICPP'05)*, pages 473–482, 2005.

[HR93] Phu Hoang and J. Rabey. Scheduling of DSP programs onto multiprocessors for maximum throughput. *IEEE Transactions on Signal Processing*, 41(6):2225–2235, 1993.

[IO98] E. Iwata and K. Olukotun. Exploiting coarse-grain parallelism in the MPEG-2 algorithm, 1998.

[IS04] ISO/IEC/JTC-1/SC-29. ISO/IEC 13818-1/Amd2:2004: Information technology - generic coding of moving pictures and associated audio information: Systems - amendment 2: Support of IPMP on MPEG-2 system, 2004.

[KA96] Yu-Kwong Kwok and Ishfaq Ahmad. Dynamic critical-path scheduling: An effective technique for allocating task graphs to multiprocessors. *IEEE Transactions on Parallel Distributed Systems*, 7(5):506–521, 1996.

[KMP06] Sang-Uok Kum and Ketan Mayer-Patel. Reference stream selection for multiple depth stream encoding. In *Third International Symposium on 3D Data Processing, Visualization, and Transmission (3DPVT'06)*, pages 623–630, 2006.

[KSJ05] Nikolay Kavaldjiev, Gerard J. M. Smit, and Pierre G. Jansen. Throughput of streaming applications running on a multiprocessor architecture. In *Proceedings of the 8th Euromicro Conference on Digital System Design (DSD '05)*, pages 350–355, 2005.

[LCM+05] Chi-Keung Luk, Robert Cohn, Robert Muth, Harish Patil, Artur Klauser, Geoff Lowney, Steven Wallace, Vijay Janapa Reddi, and Kim Hazelwood. Pin: Building customized program analysis tools with dynamic instrumentation. In *Proceedings of the 2005 ACM SIGPLAN conference on Programming language design and implementation (PLDI '05)*, pages 190–200, 2005.

[LDWL06] Shih-Wei Liao, Zhaohui Du, Gansha Wu, and Guei-Yuan Lueh. Data and computation transformations for brook streaming applications on multiprocessors. In *Proceedings of the International Symposium on Code Generation and Optimization (CGO '06)*, pages 196–207, 2006.

[LHC07] Nien-Shien Lin, Ting-Hao Huang, and Bing-Yu Chen. 3D model streaming based on JPEG 2000. *IEEE Transactions on Consumer Electronics*, 53(1):182–190, 2007.

[LLP98] M. Lee, W. Liu, and V.K. Prasanna. A mapping methodology for designing software task pipelines for embedded signal processing. In *Proceedings of the 3rd International Workshop on Embedded HPC Systems and Applications (EHPC '98)*, pages 937–944, 1998.

[LLP01] M. Lee, W. Liu, and V.K. Prasanna. Parallel implementation of a class of adaptive signal processing applications. *Algorithmica*, 30(4):645–684, 2001.

[LVK05] Ping Li, Bharadwaj Veeravalli, and Ashraf A. Kassim. Design and implementation of parallel video encoding strategies using divisible load analysis. *IEEE Transactions on Circuits and Systems for Video Technology*, 15(9):1098–1112, September 2005.

[MF01] C. A. Moritz and M. I. Frank. LoGPC: Modeling network contention in message-passing programs. *IEEE Transactions on Parallel and Distributed Systems*, 12(4):404–415, 2001.

[MLS94] B. A. Malloy, E. L. Lloyd, and M. L. Soffa. Scheduling DAG's for asynchronous multiprocessor execution. *IEEE Transactions on Parallel and Distributed Systems*, 5(5):498–508, 1994.

[MS96] Larry W. McVoy and Carl Staelin. lmbench: Portable tools for performance analysis. In *USENIX Annual Technical Conference*, pages 279–294, 1996.

[NBB06] Maik Nijhuis, Herbert Bos, and Henri E. Bal. Supporting reconfigurable parallel multimedia applications. In *Euro-Par 2006 Parallel Processing*, volume 4128 of *LNCS*, pages 765–776, 2006.

[NT93] M. G. Norman and P. Thanisch. Models of machines and computation for mapping in multicomputers. *ACM Computing Surveys*, 25(3):263–302, 1993.

[Pac] Hewlett Packard. Netperf - network performance tool. http://www..netperf.org.

[Pel98] Susanna Pelagatti. *Structured Development of Parallel Programs.* Taylor and Francis, London, 1998.

[RA01] Samantha Ranaweera and Dharma P. Agrawal. Scheduling of periodic time critical applications for pipelined execution on heterogeneous systems. In *Proceedings of the 2001 International Conference on Parallel Processing (ICPP '02)*, pages 131–140, 2001.

[RBL04] Germán Rodríguez, Rosa M. Badia, and Jesús Labarta. Generation of simple analytical models for message passing applications. In *Euro-Par 2004 Parallel Processing*, volume 3149 of *LNCS*, pages 183–188, 2004.

[RNR+03] Umakishore Ramachandran, Rishiyur S. Nikhil, James M. Rehg, Yavor Angelov, Arnab Paul, Sameer Adhikari, Kenneth M. Mackenzie, Nissim Harel, and Kathleen Knobe. Stampede: A cluster programming middleware for interactive stream-oriented applications. *IEEE Transactions on Parallel and Distributed Systems*, 14(11):1140–1154, 2003.

[RRG07] Concepció Roig, Ana Ripoll, and Fernando Guirado. A new task graph model for mapping message-passing applications (pendiente de publicación). *IEEE Transactions on Parallel Distributed Systems*, 2007.

[RSB94] S. Ramaswamy, S. Sapatnekar, and P. Banerjee. A framework for exploiting data and functional parallelism on distributed memory multicomputers, 1994.

[Sei02] D. Seinstra, F.J. Koelma. P-3PC: A point-to-point communication model for automatic and optimal decomposition of regular domain problems. *IEEE Transactions on Parallel and Distributed Systems*, 13(7):758–768, July 2002.

[SV00] Jaspal Subhlok and Gary Vondran. Optimal use of mixed task and data parallelism for pipelined computations. *Journal on Parallel and Distributed Computing*, 60(3):297–319, 2000.

[TKA02] William Thies, Michal Karczmarek, and Saman P. Amarasinghe. StreamIt: A language for streaming applications. In *Computational Complexity*, pages 179–196, 2002.

[VKS+06] Naga Vydyanathan, S. Krishnamoorthy, G. Sabin, Umit V. Catalyurek, Tahsin M. Kurc, P. Sadayappan, and Joel H. Saltz. Locality conscious processor allocation and scheduling for mixed parallel applications. In *IEEE International Conference on Cluster Computing (Cluster 2006)*, pages CD–ROM, September 2006.

[WLTC04] Eirc Hsiao-Kuang Wu, Hsu-Te Lai, Meng-Feng Tsai, and Cheng-Fu Chou. Low latency and efficient packet scheduling for streaming applications. In *IEEE International Conference on Communications (ICC'2004)*, volume 4, pages 963–1967, June 2004.

[YKS03] M-T. Yang, R. Kasturi, and A. Sivasubramaniam. A pipelined-based approach for scheduling video processing algorithms on NOW. *IEEE Transactions on Parallel and Distributed Systems*, 14(2):119–130, February 2003.

[YMW04] Hongfeng Yu, Kwan-Liu Ma, and Joel Welling. A parallel visualization pipeline for terascale earthquake simulations. In *IEEE Proceedings of the International Conference on Supercomputing (ICS '04)*, pages CD–ROM, 2004.

[ZCA06] Qian Zhu, Liang Chen, and Gagan Agrawal. Supporting a real-time distributed intrusion detection application on gates. In *Euro-Par 2006 Parallel Processing*, volume 4128 of *LNCS*, pages 360–370, 2006.

[ZFE02] Alex P. Zijdenbos, Reza Foghani, and Alan C. Evans. Automatic 'Pipeline' analysis of 3D MRI data for clinical trials: Application to multiple sclerosis. *IEEE Transactions on Medical Imaging*, 21(10):1280–1291, 2002.

[ZK07] Xinyu Zhang and Young J. Kim. Interactive collision detection for deformable models using streaming AABBs. *IEEE Transactions on Visualization and Computer Graphics*, 13(2):318–329, 2007.