Problem statement for master thesis

The master thesis will concern the design and development of a network system for use with distributed robot-controllers and similar setups.
The idea is that each separate part of the robot-controller is connected to the network through a "node". A node provides a number of registers to the robot-controller, and automatically and reliably mirrors these across the other nodes in the network. The nodes should be self-contained, and consist of a network interface, an interface for the registers, an FPGA, and the necessary glue components. An additional interface for subnodes, for example SPI to a slave-part of the robot-controller, can be implemented if time allows for it.
To avoid spending too much time on board design, an existing FPGA-board (such as a Zefant board) will be used.
During the FORK project, the possible uses of an FPGA-board with a PCI Express interface have been investigated. To build on the FORK project, the used FPGA-board will be implemented as a kind of optional master-node, with additional control- and supervision-functionality to be used through a PC system.
The main tasks of the project will be the boarddesign, and the design and implementation of a reliable network protocol and registers in an FPGA.
The requirements for the project have been divided into groups based on their importance and area, and can be seen below. The actual value of certain of the requirements will be specified during the master thesis.

High-priority requirements

These requirements are essential for the project.

1. *Functionality*
   a. A node must be able to interface with a network.
   b. Each node must provide a number of registers, and be able to mirror these across the used network.
   c. A masternode must be created using the Spartan3 PCI Express Starter kit. This has to provide various control- and supervision-functions.
2. *Components*
   a. The nodes must be based on a Xilinx FPGA.
   b. The network should be based on the TOSLINK standard, with the possibility of using electrical wiring for subnodes.
3. *Performance*
   a. Latency – a change to a register in a node must be mirrored across the network within a specified timeframe.
4. *Robustness*
   a. The network must perform some sort of error checking on the transmitted data to ensure a reliable transmission.

Medium-priority requirements

These requirements are not essential for the project, but will still be considered if time allows for it.

1. *Functionality*
    a. The nodes must support a basic command interface allowing interrupts and similar.
    b. It should be possible to attach subnodes to the nodes using for instance SPI.
2. *Robustness*
    a. The network must be able to cope with nodes being removed and added without having to be manually reset. Transient upsets are accepted for a specified timeframe after a node is plugged in / unplugged, until the network stabilizes.
3. *Physical*
    a. The total price of the components used for a node must not exceed a specified amount.

Low-priority requirements

These requirements are not important, but will be considered if time allows for it, and if the requirements do not conflict too much with the other requirements.

1. *Physical*
    a. The physical dimensions of a node may not exceed a specified size.

Tasks

From this, the following tasks can be derived:

1. *Boards to be designed:*
    a. The network-interface board for the nodes.
    b. A network-board for the Spartan3 PCI Express Starter kit masternode.
2. *Design of network, including deciding on the following parameters:*
    a. Network topology (ring, daisy chain, mesh, star, etc).
    b. Network encoding
    c. Network protocol
    d. Error-checking
3. *Implementation:*
    a. The network protocol and registers must be implemented in the FPGA.
    b. An application must be created for use with the masternode functionality.

Preliminary time-schedule

Based on the derived tasks, the following preliminary time-schedule has been drawn up:

Task	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**	**21**	**22**
Startup / Planning	X						-										
Board design		X	X				-	X									
Network design	X			X	X	X	-	X									
Implementation					X	X	-	X	X	X	X	X	X	X			
- Network protocol					X	X	-	X	X								
- Registers, etc							-		X	X							
- Masternode func.							-					X	X	X			
- Masternode app.							-						X	X			
Assembly							-	X		X							
Test							-	X	X	X	X	X	X	X	X		
Documentation							-								X	X	X