

Module 2: Stop-and-Wait Protocol

Module: Quick Look

Abstract:

Networking Infrastructure is an important concept that many high school curriculums do not cover. One of the easiest ways to present students with Networking Infrastructure concepts is to discuss ways of how data is communicated over the Internet. The purpose of this module is to provide structure for a teacher to be able to implement the Stop-and-Wait Protocol which fulfills the basic need for network infrastructure education discussed above.

Intended Audience:

- Students who have completed first semester of AP Computer Science (or equivalent)

Material Requirements:

- Programming language of instructor's choice installed on computer (suggestions for IDE in the "Module: Suggestions" section)

Total Instructor-Preparation Required:



- 60 minutes reading (assuming no familiarity with infrastructures concepts)
- 60 minutes planning

Total Class-Time Required:



- 120-200 minutes instructional time and programming time for students
- 5-10 additional minutes per student/group required to allow them to share work

Key Terms:

- | | |
|--------------------------|----------------------------------|
| • Packet | • Time frame/Timeout period |
| • End System | • ACKnowledgment (ACK) |
| • Sender | • Lost Packet |
| • Receiver | • ARQ (automatic repeat request) |
| • Stop-and-wait protocol | |

The following resources are available in the module to assist you with preparing to present the topic in class:

- **Module: Suggestions** prepares you to use the module and developing your lesson plan.
- **Module: Scope & Sequence** provides you with the objectives for your students, the basic instructional content, and ways to have students demonstrate their understanding of the protocol through a project.
- **Module: Extensions** identifies ways for you to enrich or modify the suggestion to better meet the needs of your students.

Total Instructional Time Need: 2 hours – 3.5 hours*

*based on students' programming ability more or less time may be needed

Module: Suggestions

The following are suggestions to assist you in preparing for use of this module:

- After reading through the entire module, plan your unit of study based upon your comfort with module content, your ability to program and/or teach programming, and how much available time you have in your schedule.
- If necessary, utilize the supporting materials accompanying this module (Module2_Resources.docx) to increase your understanding of the protocol. You could also provide students with this information as needed.
- Determine IDE (development environment) and programming language to be used:
 - Processing
 - <https://processing.org/>
 - Eclipse: JAVA
 - <https://eclipse.org/downloads/packages/eclipse-ide-java-developers/keplersr1>
 - Other: This module provides only suggestions so feel free to use what you like:
 - Python, Scratch, C, etc.

For this assignment, if you choose a different development environment then what you currently teach, you may need to factor in some additional instructional time.

Also, depending on your focus of graphics throughout the Computer Science course, you may need to review graphical applications including but not limited to JFrame, Graphics, and Graphics2D classes.

- Note: if you like the idea of this module but you or your students lack programming skills, the project of illustrating the Stop-and-Wait can be done in any presentation software program (such as Microsoft PowerPoint). This process would not be recommended for students who are able to program.
- Present an introductory lesson pertaining to “How the Internet Works.”
 - Lead a discussion to find out what students currently know about the Internet
 - Have students draw/illustrate their own interpretation of how the Internet works.
 - Show a video illustrating how the Internet works:
 - <http://www.warriorsofthe.net>

Module: Scope & Sequence

Suggested Timeline:

40 – 45 minutes ~ “How does the Internet work?” ~ see “Module Suggestions” ***

40 – 45 minutes ~ Presenting & Discussing the Stop-and-Wait Protocol

80 – 90 minutes ~ Coding the Project

15 – 20 minutes ~ Sharing (assumed approximately 4 groups) ***

*** if you decide to complete this portion

Teacher should present the following information to students prior to the coding project:

Learning Objectives:

Upon completion of this module, the student will be able to:

- Describe the Stop-and-Wait Protocol
- Identify the components involved and explain their role
 - sender, receiver, packet (including numbering), acknowledgment (ACK), time frame, lost packet, automatic repeat request (ARQ), and end system
- Evaluate various scenarios of Stop-and-Wait Protocol
 1. Normal operation
 2. Timeout: occurring because ACK not sent back from receiver in time
 3. Timeout: occurring because original message not sent from sender in time
- Recognize the shortcomings of Stop-and-Wait Protocol
- Produce a graphical visualization of the Stop-and-Wait Protocol*

*Students pursuing CS as a career should also simulate Sliding Window Protocol if time allows

Sequence of Instructional Content:

- Introduce the Stop-and-Wait Protocol
- Define a packet
- Differentiate between a sender and receiver
- Identify the purpose of the time frame
- Describe the numbering of the packets
- Present various scenarios:
 - Normal operation
 - Timeout: occurring because ACK not sent back from receiver in time
 - Timeout: occurring because original message not sent from sender in time
- Complete handout (StopAndWaitHandout.pptx) solution: (StopAndWaitHandoutSolution.pptx)
- Determine the shortcomings of the protocol

Guiding Questions/Statements

- Describe the purpose of the Stop-and-Wait Protocol.
Provide reliable transfer of packet within an unreliable packet delivery system
- What happens when a sender doesn't receive a packet after a certain amount of time?
A new one is sent
- Which packet number is assigned to a new packet if "packet 0" didn't receive an ACK?
packet 0
- Can you identify issue(s) with the Stop-and-Wait Protocol?
Only one outstanding packet is being sent at a time which wastes network bandwidth and it is possible that two packets with the same number could be received by the sender

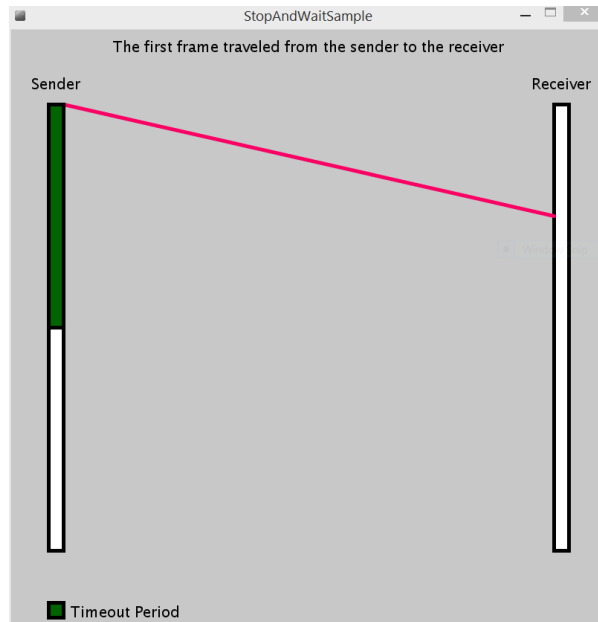
Protocol Project:

Students can program one of the following projects that illustrate all components within three scenarios of the Stop-and-Wait Protocol using the IDE and language you have chosen:

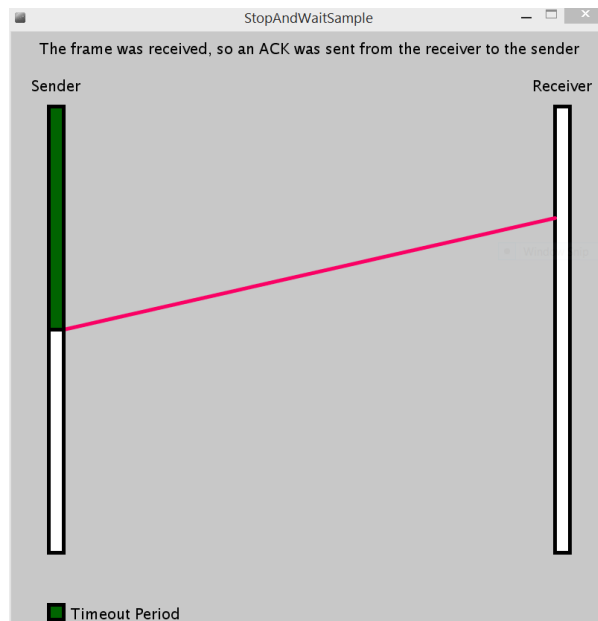
- A static diagram (similar to what is seen on the webpage resources)
- A moving diagram (similar to the included video entitled "StopAndWaitSampleProgram.wmv")
- A game based approach (students create a user-interactive game that trying to successfully deliver a packet and being required to resend packets if they are not successfully delivered)

The following is an example of possible frames for an animation project one might expect from the student played in order (programming for example below was done using Processing):

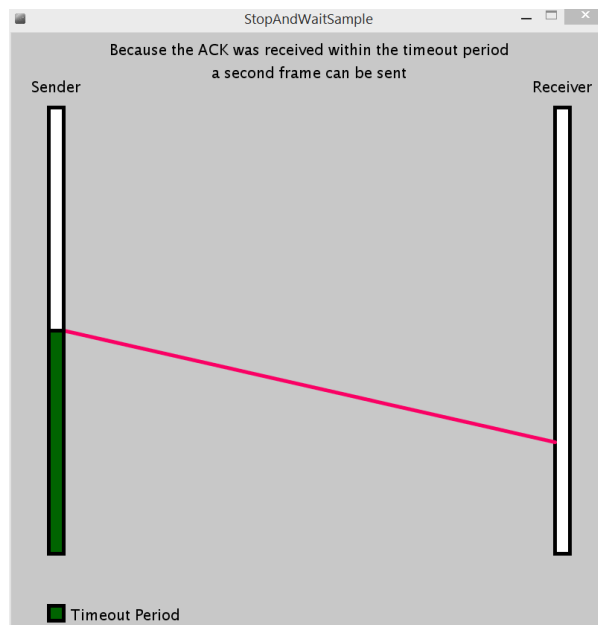
1.) The first frame traveled from the sender to the receiver



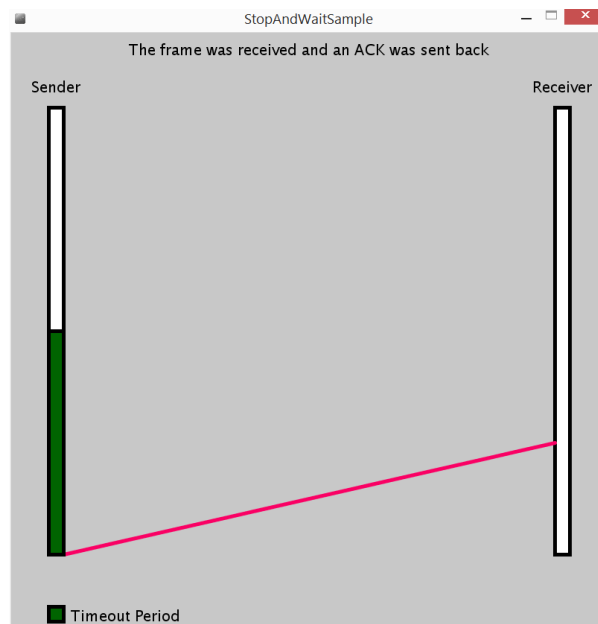
2.) The frame was received, so an ACK was sent from the receiver to the sender



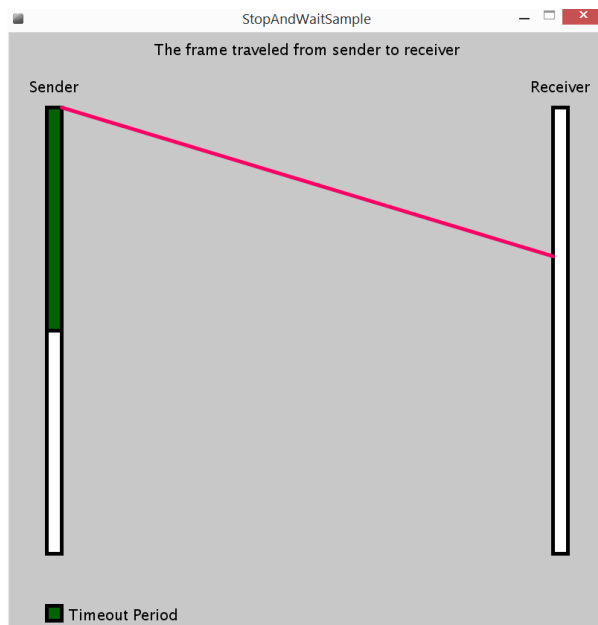
3.) Because the ACK was received within the timeout period a second frame can be sent



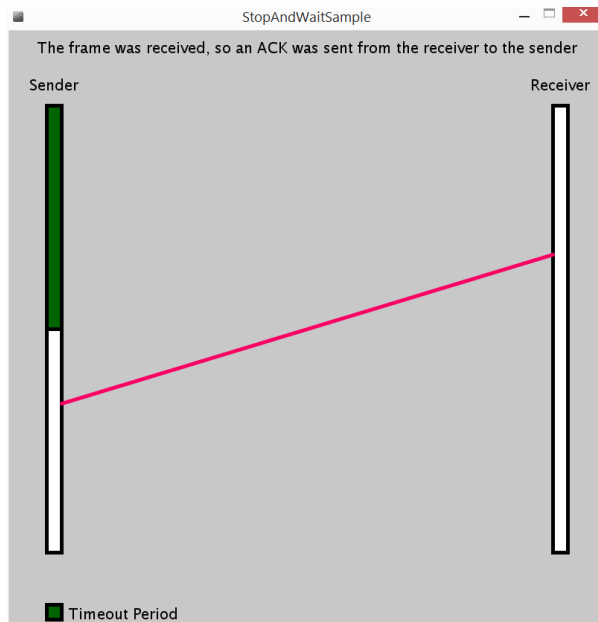
4.) The frame was received and an ACK was sent back



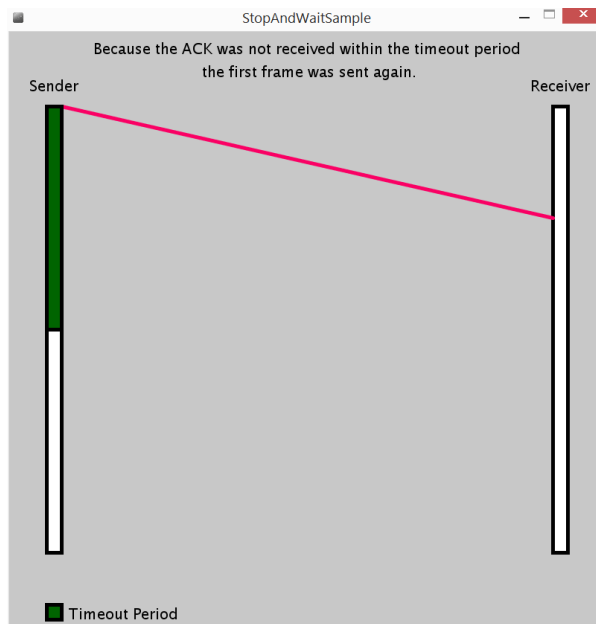
5.) The frame traveled from sender to receiver



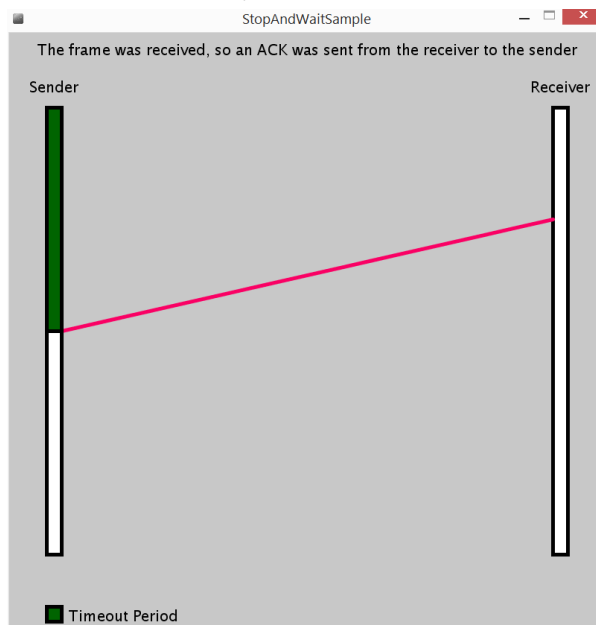
6.) The frame was received, so an ACK was sent from the receiver to the sender



7.) Because the ACK was not received within the timeout period the first frame was sent again



8.) The frame was received, so an ACK was sent from the receiver to the sender



The following assessment tool can be used to evaluate the students coding project:

This can be found under the filename: StopAndWaitRubric.xlsx

Student Name:					
Assignment Component	Earned	Possible	Comment		
Simulation Program Compiles		1		Percent Grade	0%
Illustration includes both Sender and Receiver		1			
Illustration includes sending of information from Sender to Receiver (within timeout period)		2			
Illustration includes sending of information from Sender to Receiver (not within timeout period)		2			
Illustration includes ACK sent from Receiver to Sender (within timeout period) and subsequent frame being sent		2			
Illustration includes ACK sent from Receiver to Sender (not within timeout period for any reason) and frame being resent		2			
Simulation Correctly explains (via text on screen) the steps involved in Stop-and-Wait Protocol		5			
Style Points: Visually pleasing, neatly structured code, use of methods, etc.		2			
Total Points Earned:	0	17			

Share

Even though the protocols will all be similar, students may take varying approaches in solving the illustration of the protocol. If time is available, have the students present their work to their classmates to provide them with an opportunity to share their unique approach to the project.

Module: Extensions

- Students can write an analogy comparing the Stop-and-Wait Protocol to some working system. For example: communication between two people on the hand-held radio (walkie-talkie) and how that properly/improperly demonstrates the protocol.
- Students can write a program that demonstrates programmatically, not visually, how the Stop-and-Wait protocol works. This idea would be for students who may want a little more of a challenge than creating the suggested project from above.
- Students can research and illustrate the Sliding Window Protocol using a similar approach to the steps in this module (suggested for those students pursuing CS careers).